RugFreeCoins Audit

Unified Society Quantum Token

Smart Contract Security Audit

February 28, 2022

Contents

Audit details

Audited project
Unified Society Quantum Token

Contract Address
0xea6a6467b46a8b655425ba079e8658b4ed7e2d8a

Client contact
Unified Society Quantum Team

Blockchain
Binance smart chain

Project website
www.quantumcode.org

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – please make sure to read it in full.

DISCLAIMER: By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice. No one shall have any right to rely on the report or its contents, and Rugfreecoins and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (Rugfreecoins) owe no duty of care towards you or any other person, nor does Rugfreecoins make any warranty or representation to any person on the accuracy or completeness of the report. The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and Rugfreecoins hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, Rugfreecoins hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Rugfreecoins, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report. The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.

Background

Rugfreecoins was commissioned by Unified Society Quantum Team to perform an audit of the smart contract.

https://bscscan.com/token/0xea6a6467b46a8b655425ba079e8658b4ed7e2d8a

The focus of this audit is to verify that the smart contract is secure, resilient, and working according to the specifications.

The information in this report should be used to understand the risk exposure of the smart contract, project feasibility, long-term sustainability, and as a guide to improving the security posture of the smart contract by remediating the issues that were identified.
.

About the project

Unified Society Quantum is a token built on the Binance Smart Chain that is with an innovative investment use case the main purpose of which is to seek out constant revenue sources, introducing a new cryptographic hash function which will allow modern blockchains to transition to quantum blockchains. Quantum computers have an astronomical amount of computational power compared to classic computing, and we will harness this to create the next generation of blockchains. Not only quantum-proof but quantum fast! Each transaction, purchase, and sale incur an 8% buy fee and 10% sell fee.

Tokenomics

8% fee when buying

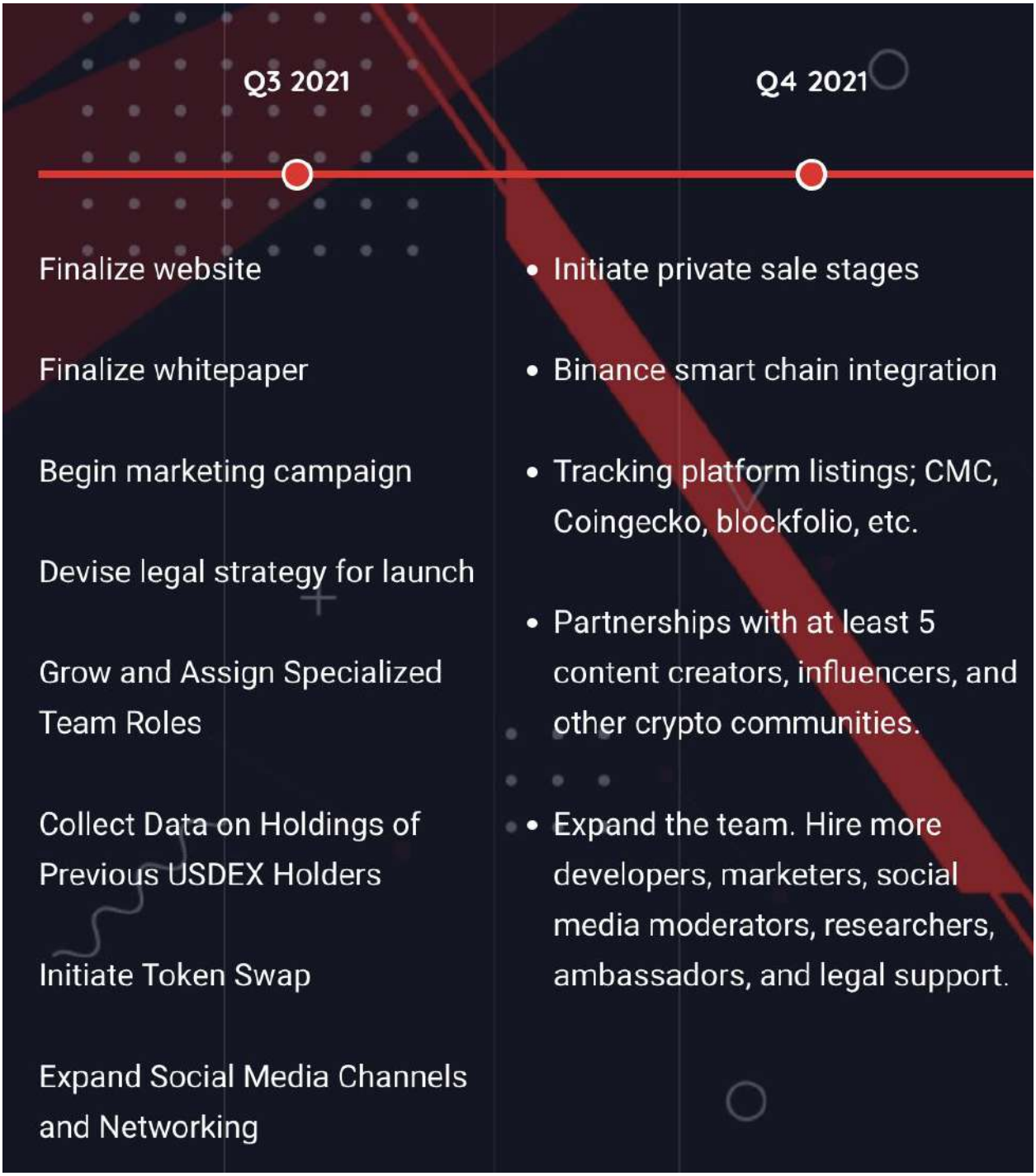
- 3% of trade goes to the marketing wallet in BNB.
- 3% of trade goes to the development wallet in BNB.
- 2% of trade goes to the buyback and burn

10% fee when selling

- Additional 2% base fee will be added when selling

2% fee in the wallet to wallet transfer

Roadmap


Q3 2021
Q4 2021
Finalize website
Finalize whitepaper
Begin marketing campaign
Devise legal strategy for launch
Grow and Assign Specialized Team Roles
Collect Data on Holdings of Previous USDEX Holders
Initiate Token Swap
Expand Social Media Channels and Networking
• Initiate private sale stages
• Binance smart chain integration
• Tracking platform listings; CMC, Coingecko, blockfolio, etc.
• Partnerships with at least 5 content creators, influencers, and other crypto communities.
• Expand the team. Hire more developers, marketers, social media moderators, researchers, ambassadors, and legal support.

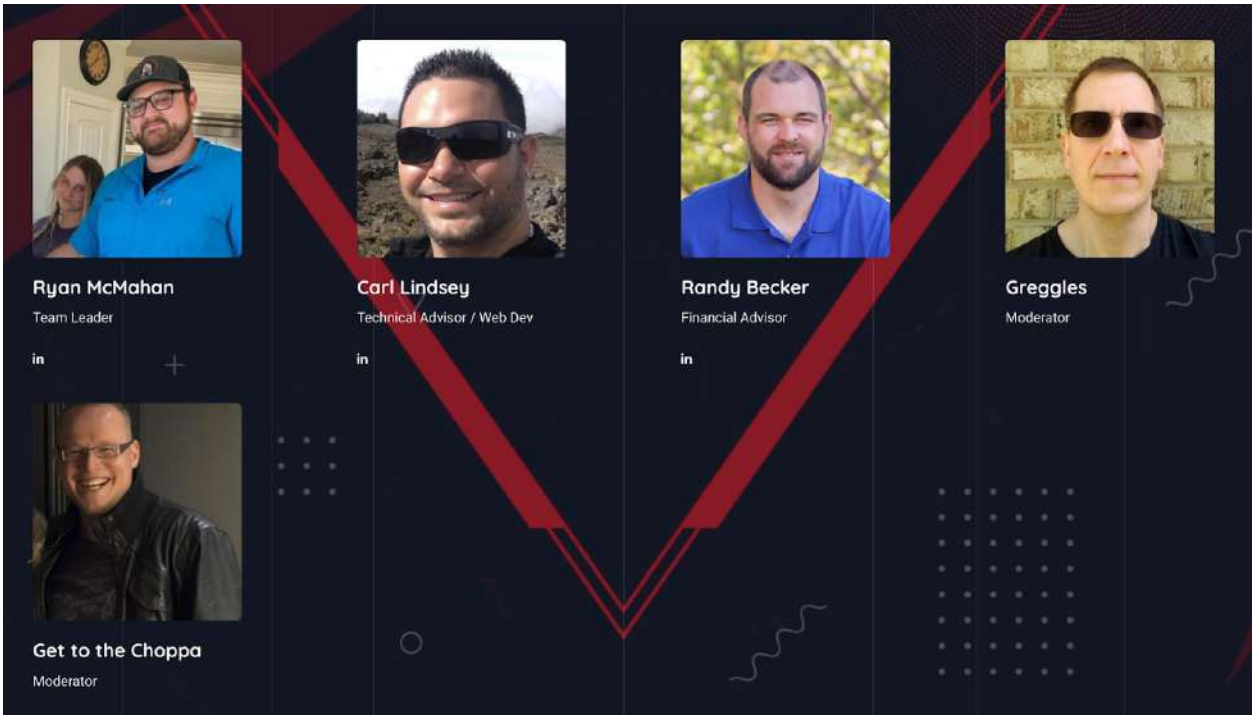
Q1 2022

- AMA videos concluding Q3 & Q4 review, accomplishments, and what lies next for Q1 & Q2.
- Initiate IDO Presale
- Top 50 exchange listing.
- Team and holders vesting schedule releases.
- Trust wallet listing.
- Pancakeswap exchange, pools, and farming listings.
- Establish partnerships with content creators, influencers, and other crypto communities.
- Continue expanding the team to at least 10.
- AMA videos with the team.
- Release new products, use cases.
- Vesting schedule releases for holders and team.
- End of year review and accomplishments.

Q2 2022

- Begin development on the premium de-fi learning platform.
- Mobile app development
- Top 20 exchange listing.
- Establish quantum research facility
- Travel the world to attend conferences, meetings, crypto events. Attend quantum physics/computer events with scientists and policy makers.

Team

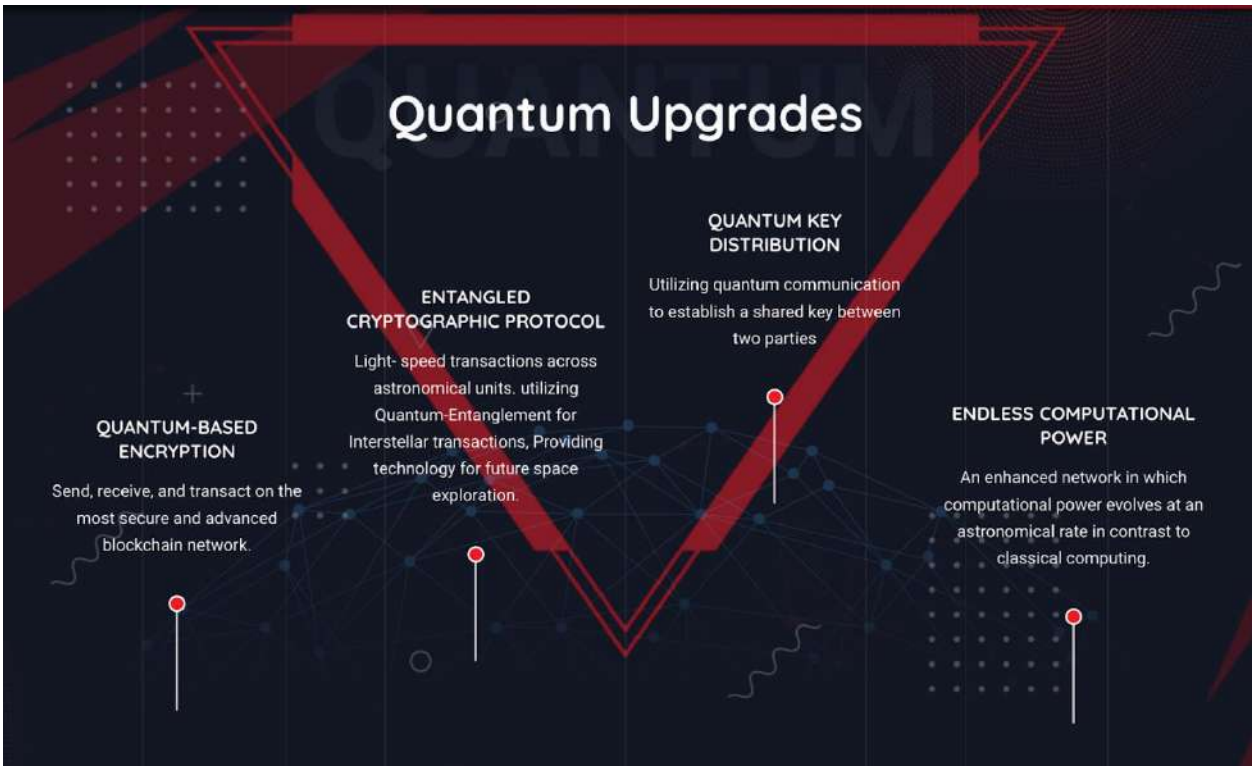

Ryan McMahan
Team Leader
in
Carl Lindsey
Technical Advisor / Web Dev
in
Randy Becker
Financial Advisor
in
Greggles
Moderator
Get to the Choppa
Moderator

Target market and the concept

Target market

- Anyone who’s interested in the Crypto space with long-term investment plans.
- Anyone who’s interested in trading tokens.
- Anyone who’s interested in taking part with the native blockchain that’s going to be built.
- Anyone who’s interested in taking part with the future plans of the Unified Society Quantum token.
- Anyone who’s interested in making financial transactions with any other party using Unified Society Quantum as the currency.

Core concept



Potential to grow with score points

1.	Project efficiency	9/10
2.	Project uniqueness	10/10
3	Information quality	10/10
4	Service quality	9/10
5	System quality	9/10
6	Impact on the community	9/10
7	Impact on the business	10/10
8	Preparing for the future	10/10
Total Points		**8.5/10**

Contract details

Token contract details for 28th February 2022

Contract name	USX Quantum
Contract address	0xeA6A6467b46A8b655425ba079e8658B4Ed7e2d8A
Token supply	2,600,000,000
Token ticker	USX
Decimals	9
Token holders	1
Transaction count	1
Contract deployer address	0x6f627be8D4f5cAf9A3fAcCC1dC1B0Fc173c98CCC
Contract's current owner address	0x6f627be8d4f5caf9a3faccc1dc1b0fc173c98ccc

Token distribution for
Tokens are distributed as follows:

Contract code function details

No	Category	Item	Result
1	Coding conventions	BRC20 Token standards	pass
		compile errors	pass
		Compiler version security	pass
		visibility specifiers	pass
		Gas consumption	pass
		SafeMath features	pass
		Fallback usage	pass
		tx.origin usage	pass
		deprecated items	pass
		Redundant code	pass
		Overriding variables	pass
2	Function call audit	Authorization of function call	pass
		Low level function (call/delegate call) security	pass
		Returned value security	pass
		Selfdestruct function security	pass
3	Business security	Access control of owners	pass
		Business logics	pass
		Business implementations	pass
4	Integer overflow/underflow		pass
5	Reentrancy		pass
6	Exceptional reachable state		pass
7	Transaction ordering dependence		pass
8	Block properties dependence		pass
9	Pseudo random number generator (PRNG)		pass
10	DoS (Denial of Service)		pass
11	Token vesting implementation		pass
12	Fake deposit		pass

13	**Event security**		**pass**

Contract description table

Below table represents the summary of the contracts and methods in the token contract. We scanned the whole contract and listed down all the Interfaces, functions, and implementations with their visibility and mutability.

Contract	**Type**	**Bases**		
└	Function Name	Visibility	Mutability	Modifiers
Context	**Implementation**			
└	_msgSender	Internal 🔒		
└	_msgData	Internal 🔒		
IERC20	**Interface**			
└	totalSupply	External ❗		NO❗
└	balanceOf	External ❗		NO❗
└	transfer	External ❗	🛑	NO❗
└	allowance	External ❗		NO❗
└	approve	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗
IERC20Metadata	**Interface**	**IERC20**		
└	name	External ❗		NO❗
└	symbol	External ❗		NO❗

└	decimals	External ❗		NO❗
ERC20	**Implementation**	**Context, IERC20, IERC20Metadata**		
└		Public ❗	🛑	NO❗
└	name	Public ❗		NO❗
└	symbol	Public ❗		NO❗
└	decimals	Public ❗		NO❗
└	totalSupply	Public ❗		NO❗
└	balanceOf	Public ❗		NO❗
└	transfer	Public ❗	🛑	NO❗
└	allowance	Public ❗		NO❗
└	approve	Public ❗	🛑	NO❗
└	transferFrom	Public ❗	🛑	NO❗
└	increaseAllowance	Public ❗	🛑	NO❗
└	decreaseAllowance	Public ❗	🛑	NO❗
└	_transfer	Internal 🔒	🛑	
└	_mint	Internal 🔒	🛑	
└	_burn	Internal 🔒	🛑	
└	_approve	Internal 🔒	🛑	

└	_beforeTokenTransfer	Internal 🔒	🛑	
Ownable	**Implementation**	**Context**		
└		Public ❗	🛑	NO❗
└	owner	Public ❗		NO❗
└	renounceOwnership	Public ❗	🛑	onlyOwner
└	transferOwnership	Public ❗	🛑	onlyOwner
IUniswapV2 Factory	**Interface**			
└	feeTo	External ❗		NO❗
└	feeToSetter	External ❗		NO❗
└	getPair	External ❗		NO❗
└	allPairs	External ❗		NO❗
└	allPairsLength	External ❗		NO❗
└	createPair	External ❗	🛑	NO❗
└	setFeeTo	External ❗	🛑	NO❗
└	setFeeToSetter	External ❗	🛑	NO❗
IUniswapV2 Pair	**Interface**			
└	name	External ❗		NO❗
└	symbol	External ❗		NO❗
└	decimals	External ❗		NO❗

└	totalSupply	External ❗		NO❗
└	balanceOf	External ❗		NO❗
└	allowance	External ❗		NO❗
└	approve	External ❗	🔘	NO❗
└	transfer	External ❗	🔘	NO❗
└	transferFrom	External ❗	🔘	NO❗
└	DOMAIN_SEPARATOR	External ❗		NO❗
└	PERMIT_TYPEHASH	External ❗		NO❗
└	nonces	External ❗		NO❗
└	permit	External ❗	🔘	NO❗
└	MINIMUM_LIQUIDITY	External ❗		NO❗
└	factory	External ❗		NO❗
└	token0	External ❗		NO❗
└	token1	External ❗		NO❗
└	getReserves	External ❗		NO❗
└	price0CumulativeLast	External ❗		NO❗
└	price1CumulativeLast	External ❗		NO❗
└	kLast	External ❗		NO❗
└	mint	External ❗	🔘	NO❗
└	burn	External ❗	🔘	NO❗
└	swap	External ❗	🔘	NO❗

└	skim	External ❗	🛑	NO❗
└	sync	External ❗	🛑	NO❗
└	initialize	External ❗	🛑	NO❗
IUniswapV2 Router01	**Interface**			
└	factory	External ❗		NO❗
└	WETH	External ❗		NO❗
└	addLiquidity	External ❗	🛑	NO❗
└	addLiquidityETH	External ❗	💵	NO❗
└	removeLiquidity	External ❗	🛑	NO❗
└	removeLiquidityETH	External ❗	🛑	NO❗
└	removeLiquidityWithPermit	External ❗	🛑	NO❗
└	removeLiquidityETHWithPermit	External ❗	🛑	NO❗
└	swapExactTokensForTokens	External ❗	🛑	NO❗
└	swapTokensForExactTokens	External ❗	🛑	NO❗
└	swapExactETHForTokens	External ❗	💵	NO❗
└	swapTokensForExactETH	External ❗	🛑	NO❗
└	swapExactTokensForETH	External ❗	🛑	NO❗
└	swapETHForExactTokens	External ❗	💵	NO❗
└	quote	External ❗		NO❗
└	getAmountOut	External ❗		NO❗

└	getAmountIn	External ❗		NO❗
└	getAmountsOut	External ❗		NO❗
└	getAmountsIn	External ❗		NO❗
IUniswapV2 Router02	**Interface**	**IUniswap V2Router 01**		
└	removeLiquidityETHSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
└	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
└	swapExactTokensForTokensSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
└	swapExactETHForTokensSupportingFeeOnTransferTokens	External ❗	💸	NO❗
└	swapExactTokensForETHSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
USXToken	**Implementation**	**ERC20, Ownable**		
└		Public ❗	🛑	ERC20
└		External ❗	💸	NO❗
└	blockTime	External ❗		NO❗
└	exchangeRouter	External ❗		NO❗
└	tokenPair	External ❗		NO❗
└	marketingWallet	External ❗		NO❗

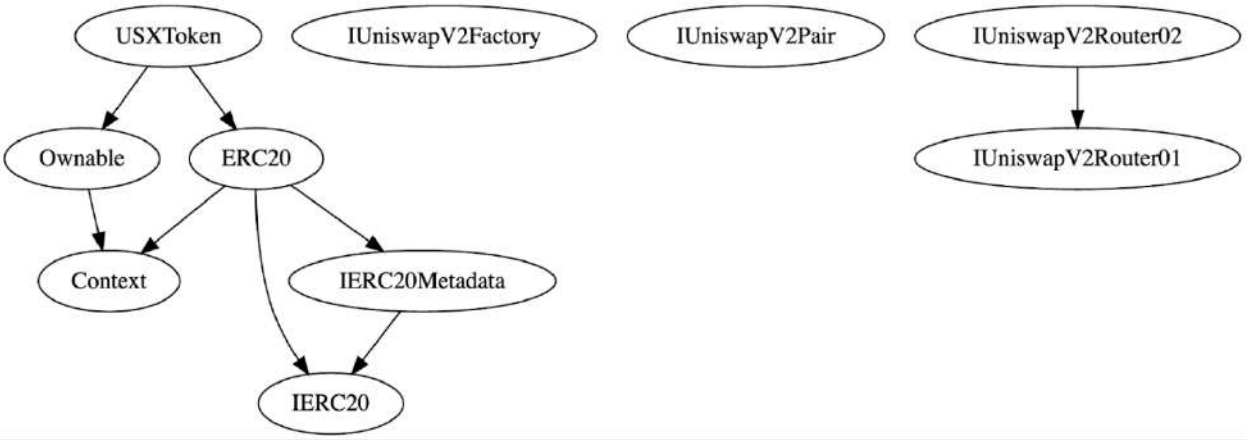
└	teamWallet	External ❗		NO❗
└	buybackWallet	External ❗		NO❗
└	walletMaxEnabled	External ❗		NO❗
└	isWalletBlackListed	External ❗		NO❗
└	isAddressLocked	External ❗		NO❗
└	isExcludedFromFees	External ❗		NO❗
└	isWalletLimitExempt	External ❗		NO❗
└	isTxLimitExempt	External ❗		NO❗
└	maxTxAmount	External ❗		NO❗
└	walletMax	External ❗		NO❗
└	swapTokensAtAmount	External ❗		NO❗
└	marketingFee	External ❗		NO❗
└	buyBackFee	External ❗		NO❗
└	baseFee	External ❗		NO❗
└	teamFee	External ❗		NO❗
└	liquityPercent	External ❗		NO❗
└	swapping	External ❗		NO❗
└	swapAndLiquifyEnabled	External ❗		NO❗
└	swapAndLiquifyByLimitOnly	External ❗		NO❗
└	updateFees	Public ❗	🛑	onlyOwner
└	changeIsTxLimitExempt	External ❗	🛑	onlyOwner

└	setMarketingWallet	External ❗	🛑	onlyOwner
└	setTeamWallet	External ❗	🛑	onlyOwner
└	setBuyBackWallet	External ❗	🛑	onlyOwner
└	_setWallet	Private 🔐	🛑	
└	updateUniswapV2Router	External ❗	🛑	onlyOwner
└	updateUniswapV2Pair	External ❗	🛑	onlyOwner
└	blacklistAddress	External ❗	🛑	onlyOwner
└	updateLockStatus	External ❗	🛑	onlyOwner
└	excludeFromFees	Public ❗	🛑	onlyOwner
└	setMaxTxAMount	External ❗	🛑	onlyOwner
└	changeWalletLimit	External ❗	🛑	onlyOwner
└	enableDisableWalletMax	External ❗	🛑	onlyOwner
└	changeIsWalletLimitExempt	External ❗	🛑	onlyOwner
└	changeSwapBackSettings	External ❗	🛑	onlyOwner
└	_transfer	Internal 🔒	🛑	
└	_swapBack	Internal 🔒	🛑	lockTheSwap
└	_addLiquidity	Private 🔐	🛑	

Legend

Symbol	Meaning
	**Function can modify state**
	**Function is payable**

Inheritance Hierarchy



Security issue checking status

- **High severity issues**

No high severity issues found

- **Medium severity issues**

No medium severity issues found

- **Low severity issues**

No low severity issues found

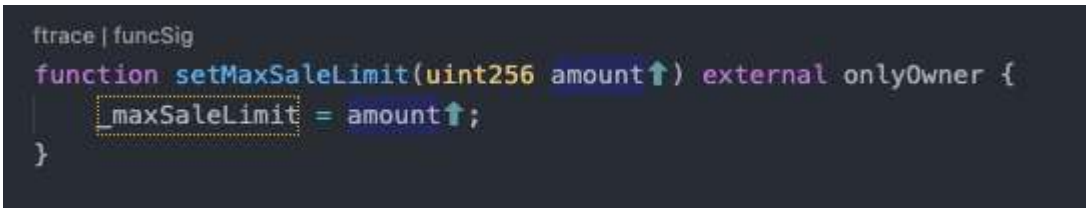
Informed and Fixed

- ❖ The owner can change all fees without any limit (can set 100%) – fix

```
ftrace | funcSig
function setAllBuyFees(
    uint256 taxFee,
    uint256 liquidityFee,
    uint256 marketingFee
) public onlyOwner {
    _taxFee = taxFee;
    _previousTaxFee = taxFee;
    _liquidityFee = liquidityFee;
    _previousLiquidityFee = liquidityFee;
    _marketingFee = marketingFee;
    _previousMarketingFee = marketingFee;
}

ftrace | funcSig
function setAllSaleFees(
    uint256 taxFee,
    uint256 liquidityFee,
    uint256 marketingFee
) public onlyOwner {
    _saleTaxFee = taxFee;
    _saleLiquidityFee = liquidityFee;
    _saleMarketingFee = marketingFee;
}
```

- ❖ The owner can change max sell limit without any limit (can set 0) - fixed



```
ftrace | funcSig
function setMaxSaleLimit(uint256 amount) external onlyOwner {
    _maxSaleLimit = amount;
}
```

Owner privileges

- ❖ The owner can change all buy and sell fees maximum up to 64%

```
ftrace | funcSig
function updateFees(
    uint256 newBaseFee⬆,
    uint256 newMarketingFee⬆,
    uint256 newTeamFee⬆,
    uint256 newBuyBackFee⬆,
    uint256 newLiquidityPercent⬆
) public onlyOwner {
    require(newBaseFee⬆ <= uint256(5), "USX: Base can not be > than 5%");
    require(
        (newMarketingFee⬆ + newTeamFee⬆ + newBuyBackFee⬆) <= uint256(10),
        "USX: Total can not be > than 10%"
    );
    require(
        newLiquidityPercent⬆ < uint256(50),
        "USX: Liquidity can not be > than 50%"
    );

    _baseFee = newBaseFee⬆;
    _liquityPercent = newLiquidityPercent⬆;

    _marketingFee = newMarketingFee⬆;
    _teamFee = newTeamFee⬆;
    _buyBackFee = newBuyBackFee⬆;

    _totalFees = _marketingFee + _teamFee + _buyBackFee;
}
```

❖ The owner can include/exclude wallets from rewards

```
ftrace | funcSig
function excludeFromReward(address account⬆) public onlyOwner {
    require(!_isExcluded[account⬆], "Account is already excluded");
    if (_rOwned[account⬆] > 0) {
        _tOwned[account⬆] = tokenFromReflection(_rOwned[account⬆]);
    }
    _isExcluded[account⬆] = true;
    _excluded.push(account⬆);
}

ftrace | funcSig
function includeInReward(address account⬆) external onlyOwner {
    require(_isExcluded[account⬆], "Account is already excluded");
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (_excluded[i] == account⬆) {
            _excluded[i] = _excluded[_excluded.length - 1];
            _tOwned[account⬆] = 0;
            _isExcluded[account⬆] = false;
            _excluded.pop();
            break;
        }
    }
}
```

❖ The owner can manually burn tokens

```
ftrace | funcSig
function manualBurn(uint256 burnAmount⬆) public onlyOwner {
    removeAllFee();
    _transferStandard(owner(), burnAddress, burnAmount⬆);
    restoreAllFee();
}
```

- The owner can include/exclude wallets from fee and whale

```
ftrace | funcSig
function setExcludedFromFee(address account⬆, bool _enabled⬆)
    public
    onlyOwner
{
    _isExcludedFromFee[account⬆] = _enabled⬆;
}

ftrace | funcSig
function setExcludedFromWhale(address account⬆, bool _enabled⬆)
    public
    onlyOwner
{
    _isExcludedFromWhale[account⬆] = _enabled⬆;
}
```

- The owner can change marketing address

```
ftrace | funcSig
function setMarketingAddress(address newWallet⬆) external onlyOwner {
    marketingWallet = payable(newWallet⬆);
}
```

- Owner can change max transaction amount minimum upto 115000

```
ftrace | funcSig
function setMaxTxAMount(uint256 amount⬆) external onlyOwner {
    require(
        amount⬆ >= uint256(115000 * (10**super.decimals())),
        "USX: Max trx must be >= 115k"
    );

    _maxTxAmount = amount⬆;
}
```

- The owner can enable/disable swapping

```
ftrace | funcSig
function setSwapAndLiquifyEnabled(bool _enabled⬆) public onlyOwner {
    swapAndLiquifyEnabled = _enabled⬆;
    emit SwapAndLiquifyEnabledUpdated(_enabled⬆);
}
```

- The owner can change max wallet limit

```
ftrace | funcSig
function setMaxWalletLimit(uint256 amount⬆) public onlyOwner {
    maxLimit = amount⬆;
}
```

- The owner can change swap back upper limit and enable/disable buy back

```
ftrace | funcSig
function setBuybackUpperLimit(uint256 swapBackLimit⬆) external onlyOwner {
    swapBackUpperLimit = swapBackLimit⬆ * 10**18;
}

ftrace | funcSig
function setBuyBackEnabled(bool _enabled⬆) public onlyOwner {
    swapBackEnabled = _enabled⬆;
    emit SwapBackEnabledUpdated(_enabled⬆);
}
```

- The owner can manually swap back tokens

```
ftrace | funcSig
function manualSwapBack(uint256 amount⬆) external onlyOwner {
    swapBack(amount⬆ * 10**15);
}
```

❖ The owner can enable/disable fees

```
ftrace | funcSig
function setFeeVariable(bool _enableFee⬆) external onlyOwner {
    enableFee = _enableFee⬆;
}
```

Audit conclusion

While conducting the audit of the Unified Society Quantum smart contract in functional-wise, it contains two low severity issues since the owner has substantial control within the ecosystem.