



# SMART CONTRACT CODE REVIEW AND SECURITY ANALYSIS REPORT



**ERA**  
\$ERA

**26/09/2022**



# TOKEN OVERVIEW

---

## Fees

- Buy fees: 8%
- Sell fees: 8%

## Fees privileges

- Can set buy fees up to 10% and sell fees up to 10%

## Ownership

- Owned

## Minting

- No mint function

## Max Tx Amount / Max Wallet Amount

- Can set max tx amount and/or wallet limitations with threshold

## Blacklist

- No blacklist function

## Other privileges

- Can exclude from fees and tx limitations
-

# TABLE OF CONTENTS

- 1 **DISCLAIMER**
- 2 **INTRODUCTION**
- 3-4 **AUDIT OVERVIEW**
- 5-7 **OWNER PRIVILEGES**
- 8 **CONCLUSION AND ANALYSIS**
- 9 **TOKEN DETAILS**
- 10 **ERA TOKEN ANALYTICS &  
TOP 10 TOKEN HOLDERS**
- 11 **TECHNICAL DISCLAIMER**



# DISCLAIMER

The information provided on this analysis document is only for general information and should not be used as a reason to invest.

FreshCoins Team will take no payment for manipulating the results of this audit.

The score and the result will stay on this project page information on our website <https://freshcoins.io>

FreshCoins Team does not guarantees that a project will not sell off team supply, or any other scam strategy ( RUG or Honeypot etc )



# INTRODUCTION

**FreshCoins** (Consultant) was contracted by **ERA** (Customer) to conduct a Smart Contract Code Review and Security Analysis.

**0x67776f14E5E9e7845e33fB0c2a3C709d259D01EB**

Network: **Ethereum (ETH)**

This report presents the findings of the security assessment of Customer's smart contract and its code review conducted on **26/09/2022**



# AUDIT OVERVIEW



Security Score  
AUDIT: PASS



Static Scan  
Automatic scanning for  
common vulnerabilities



ERC Scan  
Automatic checks for  
ERC's conformance



High



Medium



Low



Optimizations



Informational



No.	Issue description	Checking Status
1	Compiler Errors / Warnings	Passed
2	Reentrancy and Cross-function	Passed
3	Front running	Passed
4	Timestamp dependence	Passed
5	Integer Overflow and Underflow	Passed
6	Reverted DoS	Passed
7	DoS with block gas limit	Low
8	Methods execution permissions	Passed
9	Exchange rate impact	Passed
10	Malicious Event	Passed
11	Scoping and Declarations	Passed
12	Uninitialized storage pointers	Passed
13	Design Logic	Passed
14	Safe Zeppelin module	Passed

# OWNER PRIVILEGES

- Contract owner can't mint tokens after initial contract deploy
- Contract owner can't exclude an address from transactions
- Contract owner can disable transfer delay between transactions

```
function disableTransferDelay() external onlyOwner returns (bool) {  
    transferDelayEnabled = false;  
    return true;  
}
```

- Contract owner can exclude/include wallet from fees

```
function excludeFromFees(address account, bool excluded) public onlyOwner {  
    _isExcludedFromFees[account] = excluded;  
    emit ExcludeFromFees(account, excluded);  
}
```

- Contract owner can exclude/include wallet from tx limitations

```
function excludeFromMaxTransaction(address updAds, bool isEx)  
    public  
    onlyOwner  
{  
    _isExcludedMaxTransactionAmount[updAds] = isEx;  
}
```

- Contract owner can change max tx and max wallet amount (with threshold)

```
function updateMaxTxnAmount(uint256 newNum) external onlyOwner {  
    require(  
        newNum >= ((totalSupply() * 1) / 1000) / 1e18,  
        "Cannot set maxTransactionAmount lower than 0.1%"  
    );  
    maxTransactionAmount = newNum * (10**18);  
}  
  
function updateMaxWalletAmount(uint256 newNum) external onlyOwner {  
    require(  
        newNum >= ((totalSupply() * 5) / 1000) / 1e18,  
        "Cannot set maxWallet lower than 0.5%"  
    );  
    maxWallet = newNum * (10**18);  
}
```



## ● Contract owner can change swap settings

```
function updateSwapEnabled(bool enabled) external onlyOwner {
    swapEnabled = enabled;
}
```

## ● Contract owner can change buy fees up to 10% and sell fees up to 10%

```
function updateBuyFees(
    uint256 _marketingFee,
    uint256 _liquidityFee,
    uint256 _devFee
) external onlyOwner {
    buyMarketingFee = _marketingFee;
    buyLiquidityFee = _liquidityFee;
    buyDevFee = _devFee;
    buyTotalFees = buyMarketingFee + buyLiquidityFee + buyDevFee;
    require(buyTotalFees <= 10, "Must keep fees at 10% or less");
}

function updateSellFees(
    uint256 _marketingFee,
    uint256 _liquidityFee,
    uint256 _devFee
) external onlyOwner {
    sellMarketingFee = _marketingFee;
    sellLiquidityFee = _liquidityFee;
    sellDevFee = _devFee;
    sellTotalFees = sellMarketingFee + sellLiquidityFee + sellDevFee;
    require(sellTotalFees <= 10, "Must keep fees at 10% or less");
}
```

## ● Contract owner can change marketingWallet and devWallet addresses

Current values:

**marketingWallet** : 0x305C9eB6b34fe0237E4DF26499691A9D4218A235

**devWallet** : 0x4D1562483e88deac82A213a92A39d69816EB9715

```
function updateMarketingWallet(address newMarketingWallet)
    external
    onlyOwner
{
    emit marketingWalletUpdated(newMarketingWallet, marketingWallet);
    marketingWallet = newMarketingWallet;
}

function updateDevWallet(address newWallet) external onlyOwner {
    emit devWalletUpdated(newWallet, devWallet);
    devWallet = newWallet;
}
```

## ● Contract owner can renounce ownership

```
function renounceOwnership() public virtual onlyOwner {  
    _transferOwnership(address(0));  
}
```

## ● Contract owner can transfer ownership

```
function transferOwnership(address newOwner) public virtual onlyOwner {  
    require(newOwner != address(0), "Ownable: new owner is the zero address");  
    _transferOwnership(newOwner);  
}
```

### Recommendation:

The team should carefully manage the private keys of the owner's account. We strongly recommend a powerful security mechanism that will prevent a single user from accessing the contract admin functions. The risk can be prevented by temporarily locking the contract or renouncing ownership.



# CONCLUSION AND ANALYSIS



Smart Contracts within the scope were manually reviewed and analyzed with static tools.



Audit report overview contains all found security vulnerabilities and other issues in the reviewed code.



Found no HIGH issues during the first review.

# TOKEN DETAILS

## Details

Buy fees:	8%
Sell fees:	8%
Max TX:	30,000,000
Max Sell:	N/A

## Honeypot Risk

Ownership:	Owned
Blacklist:	Not detected
Modify Max TX:	Detected
Modify Max Sell:	Not detected
Disable Trading:	Not detected

## Others

Liquidity:	N/A
Holders:	Clean



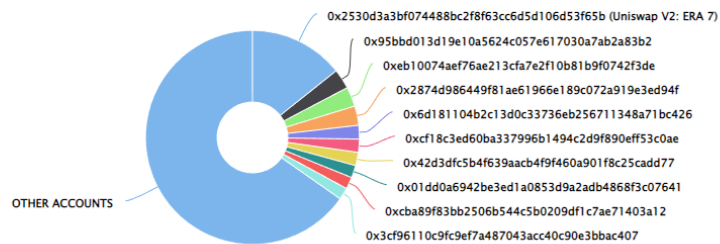
# ERA TOKEN ANALYTICS & TOP 10 TOKEN HOLDERS

The top 10 holders collectively own 34.72% (347,239,242.80 Tokens) of ERA

Token Total Supply: 1,000,000,000.00 Token | Total Token Holders: 271

ERA Top 10 Token Holders

Source: Etherscan.io



(A total of 347,239,242.80 tokens held by the top 10 accounts from the total supply of 1,000,000,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	 Uniswap V2: ERA 7	143,092,004.987500805611868228	14.3092%
2	0x95bbd013d19e10a5624c057e617030a7ab2a83b2	30,000,000	3.0000%
3	0xeb10074aef76ae213cfa7e2f10b81b9f0742f3de	29,594,974.080991057184997404	2.9595%
4	0x2874d986449f81ae61966e189c072a919e3ed94f	29,164,000.000000000584449314	2.9164%
5	0x6d181104b2c13d0c33736eb256711348a71bc426	20,655,595.280713685046667949	2.0656%
6	0xcf18c3ed60ba337996b1494c2d9f890eff53c0ae	20,000,000	2.0000%
7	0x42d3dfc5b4f639aacb4f9f460a901f8c25cadd77	20,000,000	2.0000%
8	0x01dd0a6942be3ed1a0853d9a2adb4868f3c07641	18,749,244.42883148855841267	1.8749%
9	0xcba89f83bb2506b544c5b0209df1c7ae71403a12	18,002,541.653878850655467861	1.8003%
10	0x3cf96110c9cf9ef7a487043acc40c90e3bbac407	17,980,882.365490933522949936	1.7981%

# TECHNICAL DISCLAIMER

Smart contracts are deployed and executed on the blockchain platform. The platform, its programming language, and other software related to the smart contract can have its vulnerabilities that can lead to hacks. The audit can't guarantee the explicit security of the audited project / smart contract.

