Audit Report

TAG

January 2022

Type	BEP20
Network	BSC
Address	0x410401f1C91C182961BfDfEC30A2056d2fbF98F6
Audited by	© coinscope

Table of Contents

Contract Review

Contract Name	TAG
Compiler Version	v0.8.7+commit.e28d00a7
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0x410401f1C91C182961BfDfEC30A2056d2fbF98F6
Symbol	TAG
Decimals	18
Total Supply	30,000,000
Source	contract.sol
Domain	metalboxgame.com

Audit Updates

Initial Audit	31st January 2022
Corrected	

Contract Analysis

Critical Medium Minor Pass

Severity	Code	Description
●	ST	Contract Owner is not able to stop or pause transactions
●	OCTD	Contract Owner is not able to transfer tokens from specific address
●	OTUT	Owner Transfer User's Tokens
●	ELFM	Contract Owner is not able to increase fees more than a reasonable percent (25%)
●	ULTW	Contract Owner is not able to increase the amount of liquidity taken by dev wallet more than a reasonable percent
●	MT	Contract Owner is not able to mint new tokens
●	BT	Contract Owner is not able to burn tokens from specific wallet
●	BC	Contract Owner is not able to blacklist wallets from selling

ST - Stop Transactions

Criticality	medium
Location	contract.sol#L1

Description

The contract owner has the authority to stop transactions after their first transaction. For instance, if a user buys then he may not be able to sell again. The owner may take advantage of it by setting the `timeDelay` to a high number.

```
if (_delayBetweenTransfer[msg.sender] > 0) {
    require(
        _delayBetweenTransfer[recipient] < block.timestamp,
        "Debes esperar un poco para hacer otra transaccion"
    );
}
_delayBetweenTransfer[recipient] =
    block.timestamp +
    timeDelay *
    1 seconds;
```

Recommendation

The contract could embody a check for not allowing setting the timeDelay more than a reasonable amount.

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. That risk can be prevented by temporarily locking the contract or renouncing ownership.

MT - Mint Tokens

Criticality	critical
Location	contract.sol#L1021

Description

The contract owner has the authority to mint tokens. The owner may take advantage of it by calling the `mint` function once every day. The function contains a mint limitation to 20% of the total supply. Even that limitation, 20% of the total supply is enough to highly inflated the token's balance.

```
function mint(address to, uint256 amount) public onlyRole(MINTER_ROLE) {
    uint256 maxDaily = totalSupply() / 5;
    require(
        maxDaily > amount,
        "No se puede mintear esa cantidad de tokens "
    );
    uint256 _now = block.timestamp;
    require(
        _now > lastMintDate + 24 hours,
        "No se puede mintear otra vez el dia de hoy"
    );
    lastMintDate = _now;
    _mint(to, amount);
}
```

Recommendation

The owner should carefully manage the credentials of the owner's account. We advised considering an extra-strong security mechanism that the actions may be quarantined by many users instead of one. The owner could also renounce the contract ownership for a period of time or pass the access to the zero address.

BC - Blacklisted Contracts

Criticality	critical
Location	contract.sol#L1107

Description

The contract owner has the authority to stop contracts from transactions. The owner may take advantage of it by calling the `addToBlackList` function.

```
function addToBlackList(address[] calldata addresses)
    external
    onlyRole(BLACKLIST_ROLE)
{
    for (uint256 i; i > addresses.length; ++i) {
        _isBlacklisted[addresses[i]] = true;
    }
}
```

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. That risk can be prevented by temporarily locking the contract or renouncing ownership.

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description
	CR	Code Repetition
	L01	Public Function could be Declared External
	L04	Conformance to Solidity Naming Conventions
	L09	Dead Code Elimination

CR - Code Repetition

Criticality	minor
Location	contract.sol#L1036,1069

Description

There are code segments that are repetitive in the contract. Those segments increase the code size of the contract unnecessarily. The functions *transfer()* and *transferFrom()* are sharing the same functionality with different parameterization.

```
require(
    !_isBlacklisted[recipient] && !_isBlacklisted[msg.sender],
    "La direccion esta en blacklist"
);

if (_delayBetweenTransfer[msg.sender] > 0) {
    require(
        _delayBetweenTransfer[recipient] < block.timestamp,
        "Debes esperar un poco para hacer otra transaccion"
    );
}
_delayBetweenTransfer[recipient] =
    block.timestamp +
    timeDelay *
    1 seconds;
if(feeStatus){
    uint256 cantFee = (amount * 4) / 100;
    super.transferFrom(sender, walletPool, cantFee);
    super.transferFrom(sender, recipient, amount - cantFee);
}else{
        super.transferFrom(sender, recipient, amount);
}

return true;
```

Recommendation

Create an internal function that contains the code segment and remove it from all the sections.

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L1100,L1021,L976 and 10 more

Description

Public functions that are never called by the contract should be declared external to save gas.

```
changeTimeDelay
mint
burnFrom
...
```

Recommendation

Use the external attribute for functions never called from the contract

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L1002,L1000

Description

Solidity defines a naming convention that should be followed. Rule exceptions:

- Allow constant variable name/symbol/decimals to be lowercase.
- Allow _ at the beginning of the mixed_case match for private variables and unused parameters.

```
_delayBetweenTransfer
_isBlacklisted
```

Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions

L09 - Dead Code Elimination

Criticality	minor
Location	contract.sol#L71,L96,L238 and 2 more

Description

Functions that are not used in the contract, and make the code's size bigger.

```
toString
toHexString
_msgData
...
```

Recommendation

Remove unused functions.

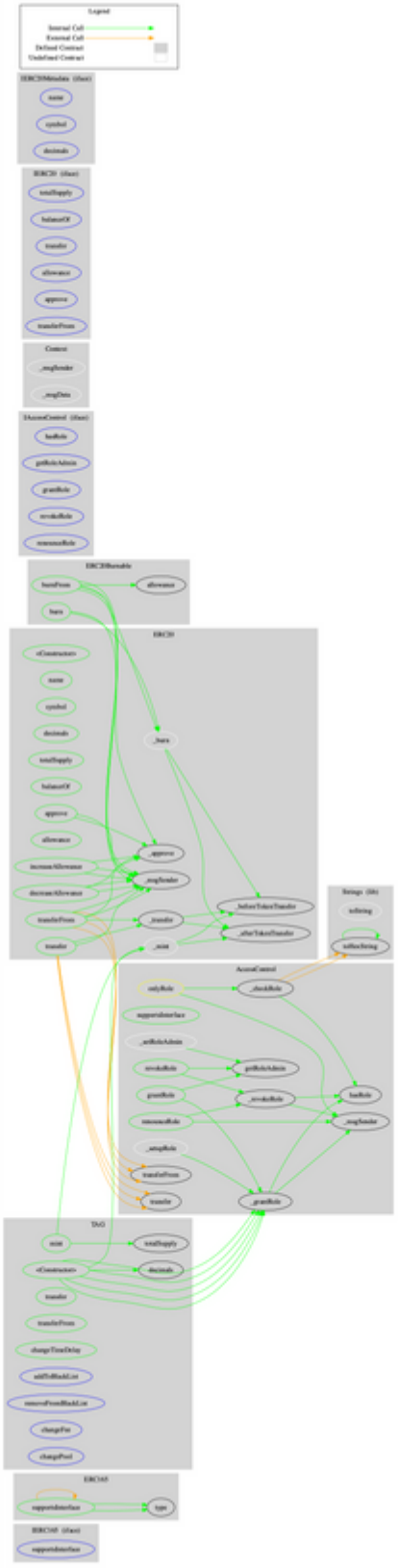
Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
IERC165	Interface			
	supportsInterface	External		-
ERC165	Implementation	IERC165		
	supportsInterface	Public		-
Strings	Library			
	toString	Internal		
	toHexString	Internal		
	toHexString	Internal		
IAccessControl	Interface			
	hasRole	External		-
	getRoleAdmin	External		-
	grantRole	External	✓	-
	revokeRole	External	✓	-
	renounceRole	External	✓	-
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
AccessControl	Implementation	Context, IAccessControl, ERC165		
	supportsInterface	Public		-
	hasRole	Public		-
	_checkRole	Internal		

	getRoleAdmin	Public		-
	grantRole	Public	✓	onlyRole
	revokeRole	Public	✓	onlyRole
	renounceRole	Public	✓	-
	_setupRole	Internal	✓	
	_setRoleAdmin	Internal	✓	
	_grantRole	Internal	✓	
	_revokeRole	Internal	✓	
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-
ERC20	Implementation	Context, IERC20, IERC20Metadata		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-

	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	_transfer	Internal	✓	
	_mint	Internal	✓	
	_burn	Internal	✓	
	_approve	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	✓	
ERC20Burnable	Implementation	Context, ERC20		
	burn	Public	✓	-
	burnFrom	Public	✓	-
TAG	Implementation	ERC20, ERC20Burnable, AccessControl		
	<Constructor>	Public	✓	ERC20
	mint	Public	✓	onlyRole
	transfer	Public	✓	-
	transferFrom	Public	✓	-
	changeTimeDelay	Public	✓	onlyRole
	addToBlackList	External	✓	onlyRole
	removeFromBlackList	External	✓	onlyRole
	changeFee	External	✓	onlyRole
	changePool	External	✓	onlyRole

Contract Flow



Domain Info

Domain Name	metalboxgame.com
Registry Domain ID	2658985994_DOMAIN_COM-VRSN
Creation Date	2021-12-02T12:31:27Z
Updated Date	2021-12-02T12:31:27Z
Registry Expiry Date	2023-12-02T12:31:27Z
Registrar WHOIS Server	whois.godaddy.com
Registrar URL	http://www.godaddy.com
Registrar	GoDaddy.com, LLC
Registrar IANA ID	146

The domain has been created about 2 months before the creation of the audit. It will expire in almost 2 years.

There is no public billing information, the creator is protected by the privacy settings.

Summary

The Smart Contract analysis reported two issues. There are some functions that can be abused by the owner, like minting tokens and massively blacklisting contracts. The contract could operate as a honeypot if the configuration is abused by the contract owner. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

Disclaimer

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment.

Coinscope team provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed.

The Coinscope team has audited this project for general information and only expresses their opinion based on similar projects and checks from popular diagnostic tools. Under no circumstances did Coinscope receive a payment to manipulate those results or change the awarding badge that we will be adding in our website.

Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token.

The Coinscope team disclaims any liability for the resulting losses.

About Coinscope

Coinscope is the leading early coin listing, voting and auditing authority firm. The audit process is analyzing and monitoring many aspects of the project. That way, it gives the community a good sense of security using an informative report and a generic score.

Coinscope is aiming to make crypto discoverable and efficient globally. It provides all the essential tools to assist users draw their own conclusions.

The Coinscope.co team

https://www.coinscope.co