



**BlockSAFU**

# **ADVANCE MANUAL SMART CONTRACT AUDIT**



**Project:** Endless Burn

**Website:** [endlessburn.com](https://endlessburn.com)



**BlockSAFU Score: 82**

**Contract Address:**

**0x4734C20B7533b5b210d0Ca66DA5055b8D7120f59**

Disclaimer: BlockSAFU is not responsible for any financial losses.  
Nothing in this contract audit is financial advice, please do your own reasearch.

## DISCLAIMER

BlockSAFU has completed this report to provide a summary of the Smart Contract functions, and any security, dependency, or cybersecurity vulnerabilities. This is often a constrained report on our discoveries based on our investigation and understanding of the current programming versions as of this report's date. To understand the full scope of our analysis, it is vital for you to at the date of this report. To understand the full scope of our analysis, you need to review the complete report. Although we have done our best in conducting our investigation and creating this report, it is vital to note that you should not depend on this report and cannot make any claim against BlockSAFU or its Subsidiaries and Team members on the premise of what has or has not been included in the report. Please remember to conduct your independent examinations before making any investment choices. We do not provide investment advice or in any way claim to determine if the project will be successful or not.

By perusing this report or any portion of it, you concur to the terms of this disclaimer. In the unlikely situation where you do not concur to the terms, you should immediately terminate reading this report, and erase and discard any duplicates of this report downloaded and/or printed by you. This report is given for data purposes as it were and on a non-reliance premise and does not constitute speculation counsel. No one should have any right to depend on the report or its substance, and BlockSAFU and its members (including holding companies, shareholders, backups, representatives, chiefs, officers, and other agents) BlockSAFU and its subsidiaries owe no obligation of care towards you or any other person, nor does BlockSAFU make any guarantee or representation to any individual on the precision or completeness of the report.

### ABOUT THE AUDITOR:

BlockSAFU (BSAFU) is an Anti-Scam Token Utility that reviews Smart Contracts and Token information to Identify Rug Pull and Honey Pot scamming activity. BlockSAFU's Development Team consists of several Smart Contract creators, Auditors Developers, and Blockchain experts. BlockSAFU provides solutions, prevents, and hunts down scammers. BSAFU is a utility token with features Audit, KYC, Token Generators, and Bounty Scammers. It will enrich the crypto ecosystem.

## OVERVIEW

**BlockSAFU was commissioned by Endless Burn to complete a Smart Contract audit. The objective of the Audit is to achieve the following:**

- Review the Project and experience and Development team
- Ensure that the Smart Contract functions are necessary and operate as intended.
- Identify any vulnerabilities in the Smart Contract code.

DISCLAIMER: This Audit is intended to inform about token Contract Risks, the result does not imply an endorsement or provide financial advice in any way, all investments are made at your own risk. (<https://blocksafu.com/>)


## SMART CONTRACT REVIEW

Token Name	Endless Burn
Token Symbol	BURNX
Token Decimal	18
Total Supply	100,000,000,000,000 BURNX
Contract Address	0x4734C20B7533b5b210d0Ca66DA5055b8D7120f59
Deployer Address	0x9e6145008c3AeAdb8025EF3866220D0eb6623434
Owner Address	0x4A59d3001c2e8ed968010e248Faafb600088Cea7
Tax Fees Buy	15%
Tax Fees Sell	15%
Gas Used for Buy	<i>will be updated after the DEX listing</i>
Gas Used for Sell	<i>will be updated after the DEX listing</i>
Contract Created	Jun-05-2022 05:35:13 AM +UTC
Initial Liquidity	<i>will be updated after the DEX listing</i>
Liquidity Status	Locked
Unlocked Date	<i>will be updated after the DEX listing</i>
Verified CA	Yes
Compiler	v0.8.9+commit.e5eed63a
Optimization	Enable with 200 runs
Sol License	MIT License
Top 5 Holders	<i>will be updated after the DEX listing</i>
Other	default evmVersion

## TAX

<b>BUY</b>	15%	<b>SELL</b>	15%
<b>Marketing Fee</b>	0%	<b>Marketing Fee</b>	0%
<b>Liquidity Fee</b>	3%	<b>Liquidity Fee</b>	3%
<b>Reflection Fee</b>	12%	<b>Reflection Fee</b>	12%

## TOP HOLDER

Rank	Address	Quantity	Percentage	Analytics
1	0x4a59d3001c2e8ed968010e248faafb600088cea7	100,000,000,000,000	100.0000% 	
<a href="#">[ Download CSV Export ]</a>				

## Team Review

The Endless Burn team has a nice website, their website is professionally built and the Smart contract is well developed, their social media is growing with over 291 people in their telegram group (count in audit date).

## OFFICIAL WEBSITE AND SOCIAL MEDIA

Website: <https://www.endlessburn.com/>

Telegram Group: <https://t.me/EndlessBurnOfficial>

Twitter: [https://twitter.com/Endless\\_Burn](https://twitter.com/Endless_Burn)

## MANUAL CODE REVIEW

● Recommendations for improvement.

recommendation for improvement.

The return value of an external transfer/transferFrom return value is checked.

Recommendation: use SafeERC20, or ensure that the transfer/transferFrom return value is checked

```
function transferFrom(  
  address sender,  
  address recipient,  
  uint256 amount  )  
  external returns (bool);
```

● Medium-risk

0 medium-risk code issues found

Should be fixed, could bring problems.

● High-Risk

0 high-risk code issues found

Must be fixed, and will bring problem.

● Critical-Risk

0 critical-risk code issues found

Must be fixed, and will bring problem.

# EXTRA NOTES SMART CONTRACT

## 1. IERC20

```
interface IERC20 {
    /**
     * @dev Returns the number of tokens in existence.
     */
    function totalSupply() external view returns (uint256);
    ...
    function balanceOf(address account) external view returns (uint256);
    ...
    function transfer(address recipient, uint256 amount) external returns (bool);
    ...
    function allowance(address owner, address spender) external view returns (uint256);
    ...
    function approve(address spender, uint256 amount) external returns (bool);
    ...
    function transferFrom(
        address sender,    address
        recipient,    uint256
        amount
    ) external returns (bool);

    /**
     * @dev Emitted when `value` tokens are moved from one account (`from`) to
     * another (`to`).
     *
     * Note that `value` may be zero.
     */
    event Transfer(address indexed from, address indexed to, uint256 value); ...
}
```

IERC20 Normal Base Template

## 2. SafeMath Contract

```
library SafeMath {
    /**
     *      @dev Returns the addition of two unsigned integers, with an overflow
     flag.
     *
     *      _Available since v3.4._
     */
    function tryAdd(uint256 a, uint256 b) internal pure returns (bool, uint256) {
unchecked {      uint256 c = a + b;      if (c < a) return (false, 0);
return (true, c);
    }
}

    /**
     *      @dev Returns the subtraction of two unsigned integers, with an
     overflow flag.
     *
     *      _Available since v3.4._
     */
    function trySub(uint256 a, uint256 b) internal pure returns (bool, uint256) {
unchecked {
        if (b > a) return (false, 0);
return (true, a - b);
    }
}

    ... function
mod(
uint256 a,
    uint256 b,
    string memory errorMessage
) internal pure returns (uint256) {
unchecked {
    require(b > 0, errorMessage);
    return a % b;
}
}
}
```



### 3. Contract Endless Burn

```
contract EndlessBurn is ERC20, Ownable {
    ...
    uint256 public maxTransactionAmount;    uint256 public
    swapTokensAtAmount;
    uint256 public maxWallet;
    ...
    function updateMaxAmount(uint256 newNum) external onlyOwner {
        require(newNum > (totalSupply() * 5 / 1000)/1e18, "Cannot
    set maxTransactionAmount lower than 0.5%");    maxTransactionAmount = newNum *
    (10**18);
    }

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

    function updateBuyFees(uint256 _marketingFee, uint256 _rewardsFee, uint256
    _liquidityFee, uint256 _buybackFee) external onlyOwner {    marketingBuyFee =
    _marketingFee;
        rewardsBuyFee = _rewardsFee;
        ...
        totalBuyFees = marketingBuyFee + rewardsBuyFee + liquidityBuyFee +
    buyBackBuyFee;
        require(totalBuyFees <= 20, "Must keep fees at 20% or less");
    }

    function updateSellFees(uint256 _marketingFee, uint256 _rewardsFee, uint256
    _liquidityFee, uint256 _buybackFee) external onlyOwner {    marketingSellFee =
    _marketingFee;
        ...
        totalSellFees = marketingSellFee + rewardsSellFee + liquiditySellFee + buyBackSellFee;
        require(totalBuyFees <= 30, "Must keep fees at 30% or less");
    }
    ...
}
```

## Endless Burn Contract

```
function updateBuyFees(uint256 _marketingFee, uint256
_rewardsFee, uint256 _liquidityFee, uint256
_buybackFee) external onlyOwner {    marketingBuyFee
= _marketingFee;
    rewardsBuyFee = _rewardsFee;
    ...
    totalBuyFees = marketingBuyFee + rewardsBuyFee +
liquidityBuyFee + buyBackBuyFee;
    require(totalBuyFees <= 20, "Must keep fees at 20% or less");
}

function updateSellFees(uint256 _marketingFee,
uint256 _rewardsFee, uint256 _liquidityFee, uint256
_buybackFee) external onlyOwner {    marketingSellFee
= _marketingFee;
    ...
    totalSellFees = marketingSellFee + rewardsSellFee +
liquiditySellFee + buyBackSellFee;
    require(totalBuyFees <= 30, "Must keep fees at 30% or less");
}
```

The owner cannot set fees over 20% for buy and 30% for sell

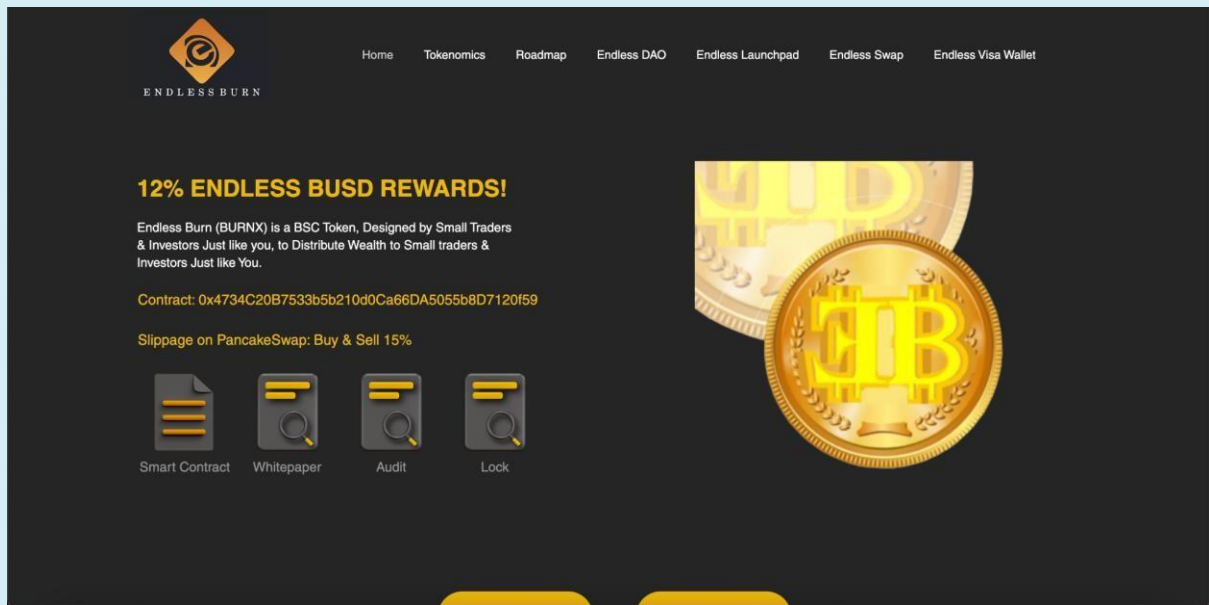
```
function updateMaxAmount(uint256 newNum) external onlyOwner
{
    require(newNum > (totalSupply() * 5 / 1000)/1e18, "Cannot
set maxTransactionAmount lower than 0.5%");
    maxTransactionAmount = newNum * (10**18);
}
```

The owner cannot set a max transaction amount lower than 0.5% of the total supply

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

The owner cannot set a max wallet amount lower than 0.5% of the total supply

## WEBSITE REVIEW



- **Mobile Friendly**
- **Contains no code error**
- **SSL Secured (By Sectigo SSL)**

**Web-Tech stack:** React, Bootstrap, Sectigo

Domain .com - Tracked by whois

First Contentful Paint:	213ms
Fully Loaded Time	5.3s
Performance	77%
Accessibility	96%
Best Practices	100%
SEO	92%

## RUG-PULL REVIEW

Based on the available information analyzed by us, we come to the following conclusions:

- Locked Liquidity (Locked by pinksale)  
*(Will be updated after DEX listing)*
- TOP 5 Holder.  
*(Will be updated after DEX listing)*
- The team is Not KYC.  
*(Will be updated after KYC)*

## HONEYPOT REVIEW

- Ability to sell.
- The owner is not able to pause the contract.
- The owner cannot set fees over 20% for buying and 30% for selling.
- The owner cannot set a max transaction amount lower than 0.5% of the total supply
- The owner cannot set a max wallet amount lower than 0.5% of the total supply

Note: Please check the disclaimer above and note, that the audit makes no statements or warranties on the business model, investment attractiveness, or code sustainability. The report is provided for the only contract mentioned in the report and does not include any other potential contracts deployed by the project owner.