



**BlockSAFU**

# **ADVANCE MANUAL SMART CONTRACT AUDIT**



**Project:** Salah ad-Din

**Website:** salah.finance



**BlockSAFU Score: 82**

**Contract Address:**

**0x18E0Eb1F63f3b1482bB9a627C97f295e7DFCe734**

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 SALAH ad-Din 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	SALAH ad-Din
Token Symbol	SALAH
Token Decimal	18
Total Supply	1,000,000,000,000 SALAH
Contract Address	0x22E12f0879EA0C7FB871AD1e58195945c79A17bE
Deployer Address	0xd2C82204bb0b0C8765c3C24CdE21FF9b2E0DfEdf
Owner Address	0x30de33f6f557acd66e7919b9379044e2de4294b6
Tax Fees Buy	10%
Tax Fees Sell	12%
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-21-2022 09:25:14 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	v0.8.4+commit.c7e474f2
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>	10%	<b>SELL</b>	12%
<b>Marketing Fee</b>	1%	Marketing Fee	1%
<b>Liquidity Fee</b>	4%	Liquidity Fee	5%
<b>Dev fee</b>	1%	Dev Fee	1%
<b>Reflection Fee</b>	4%	Reflection Fee	5%

## TOP HOLDER

Rank	Address	Quantity	Percentage	Analytics
1	0x30de33f6f557acd66e7919b9379044e2de4294b6	1,000,000,000,000	<div><div>100.0000%</div></div>	<a href="#">📊</a>

[Download CSV Export 📄]

## Team Review

The SALAH team has a nice website, their website is professionally built and the Smart contract is well developed, their social media is growing with over 416 people in their telegram group (count in audit date).

## OFFICIAL WEBSITE AND SOCIAL MEDIA

Website: <https://salah.finance/>

Telegram Group: [https://t.me/salah\\_finance](https://t.me/salah_finance)

Twitter: <https://twitter.com/SalahFinance>

## MANUAL CODE REVIEW

### ● Minor-risk

1 minor-risk code issue found

Could be fixed, and will not bring problems.

1. 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. Contract SALAH

```
contract SALAH is ERC20, Ownable {
...
    mapping (string => uint256) private buyTaxes;
    mapping (string => uint256) private sellTaxes;
    mapping (string => address) private taxWallets;
...

    // Buy Taxes
    buyTaxes["dev"] = 1;
    buyTaxes["marketing"] = 1;
    buyTaxes["liquidity"] = 4;
    buyTaxes["charity"] = 4;

    // Sell Taxes
    sellTaxes["dev"] = 1;
    sellTaxes["marketing"] = 1;
    sellTaxes["liquidity"] = 5;
    sellTaxes["charity"] = 5;

    // External wallets

    taxWallets["dev"] =
0x30de33F6F557acd66e7919B9379044e2De4294b6;
    taxWallets["charity"] =
0x9eA8eA380eaEF925406231f37c8E6AEC729B5f06;
    taxWallets["marketing"] =
0x1A51380dfE215971D375F67666504d4F4669c16B;
...

    /**
     * @dev Blacklists the specified account (Disables transfers to
and from the account).
     */
    function enableBlacklist(address account) external onlyOwner {
        require(isContract(account), "SALAH: Cannot blacklist
users");
        require(!blacklist[account], "SALAH: Account is already
blacklisted");
        blacklist[account] = true;
    }

    /**
     * @dev Remove the specified account from the blacklist.
     */
    function disableBlacklist(address account) external onlyOwner {
```



```

        require(blacklist[account], "SALAH: Account is not
blacklisted");
        blacklist[account] = false;
    }

    /**
     * @dev Sets tax for buys.
     */
    function setBuyTax(uint256 _dev, uint256 _marketing, uint256
_liquidity, uint256 _charity) external onlyOwner {

        uint totalBuyTax = _dev + _marketing + _liquidity + _charity;
        require(totalBuyTax <= 12);

        buyTaxes["dev"] = _dev;
        buyTaxes["marketing"] = _marketing;
        buyTaxes["liquidity"] = _liquidity;
        buyTaxes["charity"] = _charity;
    }

    /**
     * @dev Sets tax for sells.
     */
    function setSellTax(uint256 _dev, uint256 _marketing, uint256
_liquidity, uint256 _charity) external onlyOwner {

        uint totalSellTax = _dev + _marketing + _liquidity + _charity;
        require(totalSellTax <= 12);

        buyTaxes["dev"] = _dev;
        buyTaxes["marketing"] = _marketing;
        buyTaxes["liquidity"] = _liquidity;
        buyTaxes["charity"] = _charity;
    }
    ...

    receive() external payable {}
}

```

SALAH Contract

```

function setBuyTax(uint256 _dev, uint256 _marketing, uint256 _liquidity, uint256
_charity) external onlyOwner {
    uint totalBuyTax = _dev + _marketing + _liquidity + _charity;
    require(totalBuyTax <= 12);
    buyTaxes["dev"] = _dev;
    buyTaxes["marketing"] = _marketing;
    buyTaxes["liquidity"] = _liquidity;
    buyTaxes["charity"] = _charity;
}

```

The owner cannot set Buy Tax over 12%

```

function setSellTax(uint256 _dev, uint256 _marketing, uint256 _liquidity, uint256
_charity) external onlyOwner {
    uint totalSellTax = _dev + _marketing + _liquidity + _charity;
    require(totalSellTax <= 12);
    buyTaxes["dev"] = _dev;
    buyTaxes["marketing"] = _marketing;
    buyTaxes["liquidity"] = _liquidity;
    buyTaxes["charity"] = _charity;
}

```

The owner cannot set Sell Tax over 12%

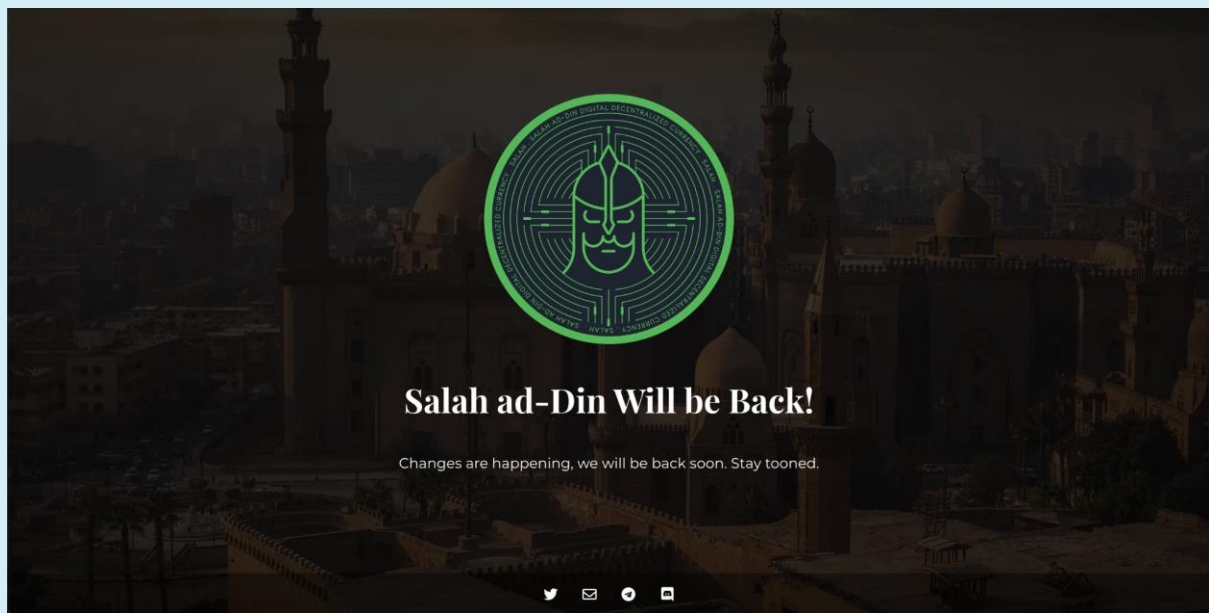
```

function enableBlacklist(address account) external onlyOwner {
    require(isContract(account), "SALAH: Cannot blacklist users");
    require(!blacklist[account], "SALAH: Account is already blacklisted");
    blacklist[account] = true;
}
/**
 * @dev Remove the specified account from the blacklist.
 */
function disableBlacklist(address account) external onlyOwner {
    require(blacklist[account], "SALAH: Account is not blacklisted");
    blacklist[account] = false;
}

```

The owner can set enable and disable the blacklisted address for the Contract

## WEBSITE REVIEW



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

**Web-Tech stack:** WordPress, Cloudflare, PHP, animate CSS

Domain .com - Tracked by whois

First Contentful Paint:	1.5s
Fully Loaded Time	3.4s
Performance	90
Accessibility	97
Best Practices	100
SEO	91

## 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 can change the fee, but can't set a fee over 10% for Buy Fees and 12% for Sell fees.

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.