



# expelee

A Secure Place For Web3

## **SMART CONTRACT AUDIT OF**

## **SIGMA Fair LAUNCH**



**Contract Address** 

0xE890AB6f84d5A1B207B29144430bA1Bf20fFA161

www.expelee.com | Page 1 |





# **Audit Summary**

Expelee team has performed a line-by-line manual analysis and automated review of the smart contract. The smart contract was analysed mainly for common smart contract vulnerabilities, exploits, and manipulation hacks. According to the smart contract audit:

**Audit Result: PASSED** 

**Ownership: NOT RENOUNCED** 

KYC Verification: Not done till date of audit

Audit Date: 04/07/2022

**Audit Team: EXPELEE** 

Be aware that smart contracts deployed on the blockchain aren't resistant to internal exploit, external vulnerability, or hack. For a detailed understanding of risk severity, source code vulnerability, functional hack, and audit disclaimer, kindly refer to the audit.

www.expelee.com | Page 2 |





# **DISCLAMER**

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. 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 Expelee 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 Expelee 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 Expelee team disclaims any liability for the resulting losses.

www.expelee.com Page 3 |





# **Contract Review**

Contract Name	AntiBotStandardToken		
Compiler Version	v0.8.4+commit.c7e474f2		
Optimization	Yes with 200 runs		
License	MIT license		
Explorer	https://bscscan.com/address/0xE890A B6f84d5A1B207B29144430bA1Bf20fFA1 61#code		
Symbol	SGM		
Decimals	18		
Total Supply	50,000,000		
Domain	https://read2earn.info/		

www.expelee.com | Page 4 |





# **Project Review**

Token Name: FitApe

Web Site: https://read2earn.info/

Twitter: https://twitter.com/ReadToEarn

Telegram: https://t.me/read2earn

**Contract Address:** 

0xE890AB6f84d5A1B207B29144430bA1Bf20fFA161

**Platform: Binance Smart Chain** 

Token Type: BEP 20

Language: SOLIDITY

www.expelee.com | Page 5 |





## **Audit Methodology**

The scope of this report is to audit the smart contract source code. We have scanned the contract and reviewed the project for common vulnerabilities, exploits, hacks, and back-doors. Below is the list of commonly known smart contract vulnerabilities, exploits, and hacks:

## Category

- Unhandled Exceptions

- Transaction Order Dependency

Smart Contract Vulnerabilities

- Integer Overflow

- Unrestricted Action

Incorrect Inheritance Order

- Typographical Errors

- Requirement Violation

Source Code Review

- Gas Limit and Loops

- Deployment Consistency

- Repository Consistency

- Data Consistency

- Token Supply Manipulation

Functional Assessment - Operations Trail & Event Generation

- Assets Manipulation

- Liquidity Access

www.expelee.com | Page 6 |



# **Vulnerability Checklist**

Νō	Description.	Result
1	Compiler warnings.	Passed
2	Race conditions and Re-entrancy. Cross-function raceconditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Front running.	Passed
6	Timestamp dependence.	Passed
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Passed
10	Methods execution permissions.	Passed
11	Economy model.	Passed
12	The impact of the exchange rate on the logic.	Passed
13	Private user data leaks.	Passed
14	Malicious Event log.	Passed
15	Scoping and Declarations.	Passed
16	Uninitialized storage pointers.	Passed
17	Arithmetic accuracy.	Passed
18	Design Logic.	Passed
19	Cross-function race conditions.	Passed
20	Safe Zeppelin module.	Passed
21	Fallback function security.	Passed

www.expelee.com | Page 7 |



## **Manual Audit**

- Low-Risk3 low-risk code issues found
- Medium-Risk0 medium-risk code issues found
  - High-Risk

    O high-risk code issues found

www.expelee.com | Page 8 |





#### 1) Contract contains Reentrancy vulnuerabilities

```
function _transfer(
       address sender,
        address recipient,
        uint256 amount
   ) internal virtual {
        require(sender != address(0), "ERC20: transfer from the zero address");
        require(recipient != address(0), "ERC20: transfer to the zero address");
        if (enableAntiBot) {
            pinkAntiBot.onPreTransferCheck(sender, recipient, amount);
        }
       _beforeTokenTransfer(sender, recipient, amount);
        _balances[sender] = _balances[sender].sub(
            amount,
            "ERC20: transfer amount exceeds balance"
       _balances[recipient] = _balances[recipient].add(amount);
       emit Transfer(sender, recipient, amount);
   }
```

#### Recommendation

Apply the check-effects-interaction pattern

www.expelee.com Page 9 |





#### 2) No zero address validation for some functions

Detect missing zero address validation.

Detect missing zero address validation.

address serviceFeeReceiver\_,
payable(serviceFeeReceiver\_).transfer(serviceFee\_)

#### Recommendation

Check that the new address is not zero.

www.expelee.com | Page 10 |





#### 3) Local Variable Shadowing

Detection of shadowing using local variables.

```
function allowance(address owner, address spender)
    public
    view
    virtual
    override
    returns (uint256)
    {
       return _allowances[owner][spender];
    }
}
```

#### Recommendation

Rename the local variables that shadow another component.

www.expelee.com Page 11 |





## **Audit Summary**

Compiled with solc

Number of lines: 836 (+ 0 in dependencies, + 0 in tests)

Number of assembly lines: 0

Number of contracts: 7 (+ 0 in dependencies, + 0 tests)

Number of optimization issues: 13 Number of informational issues: 17

Number of low issues: 7 Number of medium issues: 0 Number of high issues: 0

ERCs: ERC20

+	+			+	+
Name	# functions	ERCS	ERC20 info	Complex code	Features
+	+	t	<del>-</del>	+	+
SafeMath	13			No	
IPinkAntiBot	2			No	
AntiBotStandardToken	33	ERC20	No Minting	No	Receive ETH
1			Approve Race Cond.		Send ETH
1					
+	+	·	+	+	+

www.expelee.com Page 12 |





# Manual Audit (Contract Function)

```
contract AntiBotStandardToken is IERC20, Ownable, BaseToken {
  using SafeMath for uint256;
  uint256 public constant VERSION = 1;
  mapping(address => uint256) private _balances;
  mapping(address => mapping(address => uint256)) private _allowances;
  string private _name;
  string private _symbol;
  uint8 private decimals;
  uint256 private _totalSupply;
  IPinkAntiBot public pinkAntiBot;
  bool public enableAntiBot;
  constructor(
    string memory name_,
    string memory symbol_,
    uint8 decimals_,
    uint256 totalSupply,
    address pinkAntiBot,
    address serviceFeeReceiver,
    uint256 serviceFee
  ) payable {
    _name = name_;
    _symbol = symbol_;
    decimals = decimals_;
    _mint(owner(), totalSupply_);
    pinkAntiBot = IPinkAntiBot(pinkAntiBot );
    pinkAntiBot.setTokenOwner(owner());
    enableAntiBot = true;
    emit TokenCreated(
      owner(),
      address(this),
      TokenType.antiBotStandard,
      VERSION
    );
```

www.expelee.com | Page 13 |





## Important Points To Consider

✓ Verified contract source

X Token is sellable (not a honeypot) at this time

X Ownership renounced or source does not contain an owner contract

✓ Owner/creator wallet contains less than 10% of circulating token supply (0.46%)

www.expelee.com | Page 14 |





# About Expelee

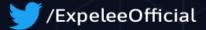
Expelee is a community driven organisation dedicated to fostering an antirug movement. We're here to keep investment safe from fraudsters. We've encountered several rug pulls and know how it feels to be duped, which is why we don't want anybody else to go through the same experience. We are here to raise awareness through our services so that the future of cryptocurrency can be rug-free.

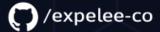
The auditing process focuses to the following considerations with collaboration of an expert team:

- Functionality test of the Smart Contract to determine if proper logic has been followed throughout the whole process.
- Manually detailed examination of the code line by line by experts.
- Live test by multiple clients using Test net.
- Analysing failure preparations to check how the Smart
- Contract performs in case of any bugs and vulnerabilities.
- Checking whether all the libraries used in the code are on the latest version.
- Analysing the security of the on-chain data.

### Social Media







www.expelee.com | Page 15 |