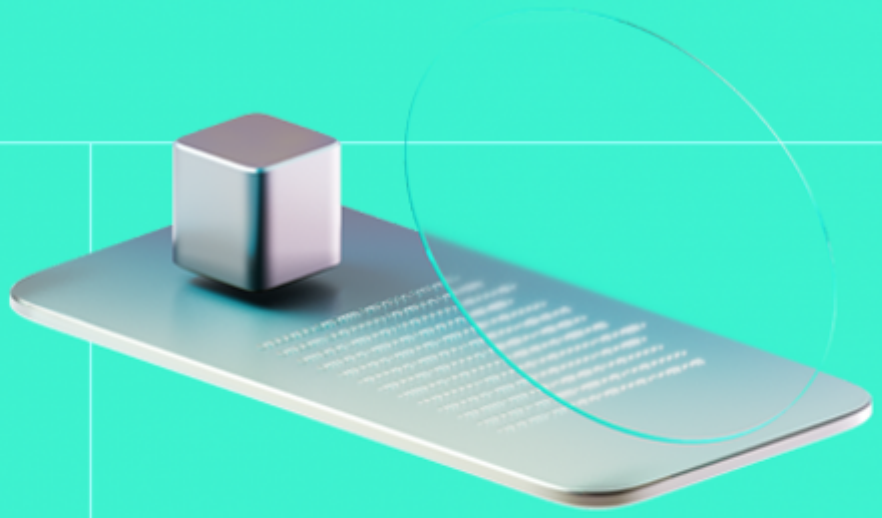
HACKEN

Smart Contract Code Review And Security Analysis Report



Customer: FBS Markets

Date: 22/06/2025

We express our gratitude to the FBS Markets team for the collaborative engagement that enabled the execution of this Smart Contract Security Assessment.

FBS Markets is Asian’s largest t inernational forex CFD and crypto broker with over 2 million registered users and a team of 350+ members that meet all KYC and AML requirements.

Platform: EVM

Language: Solidity

Timeline: 18/04/2025 - 19/06/2025

Methodology: https://hackenio.cc/sc_methodology

Audit Summary

10/10	7/10	10/10	10/10
Security Score	Code quality score	Architecture quality score	Documentation quality score

Total 10/10

The system users should acknowledge all the risks summed up in the risks section of the report

7	7	0	0
Total Findings	Resolved	Accepted	Mitigated

Findings by severity

Critical	2
High	0
Medium	0
Low	5

Vulnerability	Status
F-2022-1526 - Mint is not limited	Fixed
F-2022-1527 - Incorrect TRC20 interface	Fixed
F-2022-1528 - Variable Shadowing	Fixed
F-2022-1529 - Boolean equality	Fixed
F-2022-1531 - Zero address is allowed	Fixed
F-2022-1532 - DestroyBlackFunds event has an incorrect burned amount value	Fixed
F-2024-1530 - The public function could be declared external	Fixed

This report may contain confidential information about IT systems and the intellectual property of the Customer, as well as information about potential vulnerabilities and methods of their exploitation.

The report can be disclosed publicly after prior consent by another Party. Any subsequent publication of this report shall be without mandatory consent.

Document

Name	Smart Contract Code Review and Security Analysis Report for FBS Markets
Audited By	Hacken
Website	https://fbscoin.digital
Changelog	22/04/2025 - Preliminary Report
	03/05/2025 - Second Review
	06/05/2025 - Third Review
	19/06/2025 - Fourth Review

Table of Contents

System Overview

FBS Markets is Asian's largest t inernational forex CFD and crypto broker with over 2 million registered users and a team of 350+ members that meet all KYC and AML requirements.

Token — simple TRC-20 token that mints nothing after the deployment. Additional minting is allowed. The token has the ability to add addresses to the black list, which will stop all operations with the address. For blacklisted addresses, it has the ability to destroy funds.

It has the following attributes:

- Name: FBS Markets FBS
- Symbol: FBS
- Decimals: 8
- Total supply: 20m (20m for TRC netwrok)

Privileged roles

- The owner of the FBS contract can add or remove addresses from the blacklist to lock funds.
- The owner of the FBS contract can destroy funds for any blacklisted address.
- The owner of the FBS contract has the ability to burn tokens.
- The owner of the FBS contract can pause the contract, so all transfers would be stopped.
- The owner of the FBS contract can mint tokens without any limits.

Executive Summary

This report presents an in-depth analysis and scoring of the customer's smart contract project. Detailed scoring criteria can be referenced in the scoring methodology.

Documentation quality

The total Documentation Quality score is **10** out of **10**.

- The Customer provided superficial functional requirements.
- Technical description is not provided.

Code quality

The total Code Quality score is **7** out of **10**.

- Limited number of unit tests provided.

Architecture quality

The architecture quality score is **10** out of **10**.

- Code is well-structured and easy-readable.

Security score

Upon auditing, the code was found to contain **2** critical, **0** high, **0** medium, and **5** low severity issues, leading to a security score of **10** out of **10**.

All identified issues are detailed in the “Findings” section of this report.

Summary

The comprehensive audit of the customer's smart contract yields an overall score of **10**. This score reflects the combined evaluation of documentation, code quality, test coverage, and security aspects of the project.

Findings

Vulnerability Details

F-2022-1526 - Mint is not limited - Critical

Description: The mint function is not limited, and any amount of tokens could be minted by the owner. It breaks the requirements to have a limit of 20 million tokens.

Status: Fixed

Classification

Severity: Critical

Recommendations

Remediation: Restrict minting according to specification.

F-2022-1527 - Incorrect TRC20 interface - Critical

Description: Incorrect return values for TRC20 functions. A contract compiled with Solidity > 0.4.22 interacting with these functions will fail to execute them, as the return value is missing.

`Token.transfer` does not return a boolean. Bob deploys the token. Alice creates a contract that interacts with it but assumes a correct TRC20 interface implementation. Alice's contract is unable to interact with Bob's contract.

Status: Fixed

Classification

Severity: Critical

Recommendations

Remediation: Set the appropriate return values and types for the defined TRC20 functions.

F-2022-1528 - Variable Shadowing - Low

Description: Solidity allows for ambiguous naming of state variables when inheritance is used. Contract A with a variable x could inherit contract B, which has a state variable x defined. This would result in two separate versions of x, accessed from contract A and the other from contract B. In more complex contract systems, this condition could go unnoticed and subsequently lead to security issues.

Assets:

- TRC20.sol [https://github.com/fbs-markets/fbs-token]

Status: Fixed

Classification

Severity: Low

Recommendations

Remediation: Consider renaming function argument.

F-2022-1529 - Boolean equality - Low

Description: Boolean constants can be used directly and do not need to be compared to true or false.

Assets:

- TRC20.sol [https://github.com/fbs-markets/fbs-token]

Status: Fixed

Classification

Severity: Low

Recommendations

Remediation: Consider using “whenNotPaused” modifier instead.

F-2022-1531 - Zero address is allowed - Low

Description: `msg.sender` does not check if it is a zero address, which should be checked.

Assets:

- TRC20.sol [https://github.com/fbs-markets/fbs-token]

Status: Fixed

Classification

Severity: Low

Recommendations

Remediation: Add check for zero address for `msg.sender`.

F-2022-1532 - DestroyBlackFunds event has an incorrect burned amount value - Low

Description: `DestroyedBlackFund` emits an event with the value from user balances `_balances[_blackListedUser]` which was reset to zero a step before.

Assets:

- TRC20.sol [https://github.com/fbs-markets/fbs-token]

Status: Fixed

Classification

Severity: Low

Recommendations

Remediation: Use the `dirtyFunds` variable instead of `_balances[_blackListedUser]`.

F-2024-1530 - The public function could be declared external - Low

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

Assets:

- Ownable.sol [https://github.com/fbs-markets/fbs-token]
- Pausable.sol [https://github.com/fbs-markets/fbs-token]
- TRC20Detailed.sol [https://github.com/fbs-markets/fbs-token]
- TRC20.sol [https://github.com/fbs-markets/fbs-token]

Status: Fixed

Classification

Severity: Low

Recommendations

Remediation: Use the external attribute for functions never called from the contract.

Disclaimers

Hacken Disclaimer

The smart contracts given for audit have been analyzed based on best industry practices at the time of the writing of this report, with cybersecurity vulnerabilities and issues in smart contract source code, the details of which are disclosed in this report (Source Code); the Source Code compilation, deployment, and functionality (performing the intended functions).

The report contains no statements or warranties on the identification of all vulnerabilities and security of the code. The report covers the code submitted and reviewed, so it may not be relevant after any modifications. Do not consider this report as a final and sufficient assessment regarding the utility and safety of the code, bug-free status, or any other contract statements.

While we have done our best in conducting the analysis and producing this report, it is important to note that you should not rely on this report only — we recommend proceeding with several independent audits and a public bug bounty program to ensure the security of smart contracts.

English is the original language of the report. The Consultant is not responsible for the correctness of the translated versions.

Technical Disclaimer

Smart contracts are deployed and executed on a blockchain platform. The platform, its programming language, and other software related to the smart contract can have vulnerabilities that can lead to hacks. Thus, the Consultant cannot guarantee the explicit security of the audited smart contracts.

Appendix 1. Severity Definitions

When auditing smart contracts, Hacken is using a risk-based approach that considers **Likelihood**, **Impact**, **Exploitability** and **Complexity** metrics to evaluate findings and score severities.

Reference on how risk scoring is done is available through the repository in our Github organization:

hknio/severity-formula

Severity	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to the loss of user funds or contract state manipulation.
High	High vulnerabilities are usually harder to exploit, requiring specific conditions, or have a more limited scope, but can still lead to the loss of user funds or contract state manipulation.
Medium	Medium vulnerabilities are usually limited to state manipulations and, in most cases, cannot lead to asset loss. Contradictions and requirements violations. Major deviations from best practices are also in this category.
Low	Major deviations from best practices or major Gas inefficiency. These issues will not have a significant impact on code execution, do not affect security score but can affect code quality score.

Appendix 2. Scope

The scope of the project includes the following smart contracts from the provided repository:

Scope Details

Repository	https://github.com/fbs-markets/fbs-token
Commit	2ab92561d118bcf0801adeab5cddea86d813b61b
Requirements	Provided
Technical Requirements	Not Provided

Contracts in Scope

BlackList.sol

TRC20Detailed.sol

TRC20.sol

ITRC20.sol

Ownable.sol

Pausable.sol

SafeMath.sol

FBSToken.sol