verichains

SECURITY AUDIT OF

GVSTOKEN SMART CONTRACT

Public Report

Sep 6, 2023

Verichains Lab

info@verichains.io

https://www.verichains.io

Driving Technology > Forward

ABBREVIATIONS

Name	Description
Ethereum	An open source platform based on blockchain technology to create and distribute smart contracts and decentralized applications.
Ether (ETH)	A cryptocurrency whose blockchain is generated by the Ethereum platform. Ether is used for payment of transactions and computing services in the Ethereum network.
Smart contract	A computer protocol intended to digitally facilitate, verify or enforce the negotiation or performance of a contract.
Solidity	A contract-oriented, high-level language for implementing smart contracts for the Ethereum platform.
Solc	A compiler for Solidity.
ERC20	ERC20 (BEP20 in Binance Smart Chain or *x*RP20 in other chains) tokens are blockchain-based assets that have value and can be sent and received. The primary difference with the primary coin is that instead of running on their own blockchain, ERC20 tokens are issued on a network that supports smart contracts such as Ethereum or Binance Smart Chain.

EXECUTIVE SUMMARY

This Security Audit Report was prepared by Verichains Lab on Sep 6, 2023. We would like to thank the GVSToken for trusting Verichains Lab in auditing smart contracts. Delivering high-quality audits is always our top priority.

This audit focused on identifying security flaws in code and the design of the GVSToken Smart Contract. The scope of the audit is limited to the source code files provided to Verichains. Verichains Lab completed the assessment using manual, static, and dynamic analysis techniques.

During the audit process, the audit team had identified no vulnerability issue in the contract code.

TABLE OF CONTENTS

1. MANAGEMENT SUMMARY

1.1. About GVSToken Smart Contract

GVS Digital Asset Guarantee is a leading multinational company dedicated to providing comprehensive insurance solutions for a diverse range of assets, with a particular focus on digital assets such as NFTs and cryptocurrencies. Their mission is to safeguard the value of your investments and empower you to navigate the dynamic landscape of digital assets with confidence.

1.2. Audit scope

This audit focused on identifying security flaws in code and the design of the GVSToken Smart Contract.

The audited contract is the GVSToken Smart Contract that deployed on Binance Smart Chain Mainnet at address `0x9f68f166301ACA720aa5B5588329230d3316e358`. The details of the deployed smart contract are listed in Table 1.

FIELD	VALUE
Contract Name	GVSToken
Contract Address	0x9f68f166301ACA720aa5B5588329230d3316e358
Compiler Version	v0.8.9+commit.e5eed63a
Optimization Enabled	Yes with 200 runs
Explorer	*https://bscscan.com/address/0x9f68f166301ACA720aa5B5588329230d3316e358*

Table 1. The deployed smart contract details

1.3. Audit methodology

Our security audit process for smart contract includes two steps:

- Smart contract codes are scanned/tested for commonly known and more specific vulnerabilities using public and RK87, our in-house smart contract security analysis tool.

- Manual audit of the codes for security issues. The contracts are manually analyzed to look for any potential problems.

Following is the list of commonly known vulnerabilities that were considered during the audit of the smart contract:

- Integer Overflow and Underflow
- Timestamp Dependence
- Race Conditions
- Transaction-Ordering Dependence
- DoS with (Unexpected) revert
- DoS with Block Gas Limit
- Gas Usage, Gas Limit and Loops
- Redundant fallback function
- Unsafe type Inference
- Reentrancy
- Explicit visibility of functions state variables (external, internal, private and public)
- Logic Flaws

For vulnerabilities, we categorize the findings into categories as listed in table below, depending on their severity level:

SEVERITY LEVEL	DESCRIPTION
CRITICAL	A vulnerability that can disrupt the contract functioning; creates a critical risk to the contract; required to be fixed immediately.
HIGH	A vulnerability that could affect the desired outcome of executing the contract with high impact; needs to be fixed with high priority.
MEDIUM	A vulnerability that could affect the desired outcome of executing the contract with medium impact in a specific scenario; needs to be fixed.
LOW	An issue that does not have a significant impact, can be considered as less important.

Table 2. Severity levels

1.4. Disclaimer

Please note that security auditing cannot uncover all existing vulnerabilities, and even an audit in which no vulnerabilities are found is not a guarantee for a 100% secure smart contract. However, auditing allows discovering vulnerabilities that were unobserved, overlooked during development and areas where additional security measures are necessary.

2. AUDIT RESULT

2.1. Overview

The GVSToken Smart Contract was written in `Solidity` language, with the required version to be `^0.8.9`. The source code was written based on OpenZeppelin's library.

2.1.1. Token contract

The contract extends `ERC20`, `Ownable` and `Pausable` contracts. With `Ownable`, by default, the token Owner is contract deployer, but he can transfer ownership to another address at any time. The token Owner can pause/unpause contract using `Pausable` contract. Users, except those on the whitelist, can only transfer and receive tokens when the contract is not paused. The token Owner can add or remove addresses from the whitelist.

The contract has a blacklist mechanism to prevent the blacklisted addresses from transferring and receiving tokens. The token Owner can add or remove addresses from the blacklist.

The token owner can withdraw all native tokens and any other tokens present in the contract.

The contract will pre-mint all 500 million tokens to the `_ownerAddress`, add that address to the whitelist, and then transfer ownership to that address.

This table lists some properties of the FBB ERC20 token contract (as of the report writing time).

PROPERTY	VALUE
Name	GVS Token
Symbol	GVS
Decimals	18
Max Supply	500,000,000 (x10^{18}) Note: the number of decimals is 18, so the total representation token will be 500,000,000 or 500 million.

Table 3. The GVSToken Smart Contract properties

2.2. Findings

During the audit process, the audit team found no vulnerability in the given version of GVSToken Smart Contract.

2.3. Additional notes and recommendations

2.3.1. Lack of necessary events INFORMATIVE

Events are used to inform external users that something happened on the blockchain. Smart contracts themselves cannot listen to any events.

All information in the blockchain is public and any actions can be found by looking into the transactions close enough but events are a shortcut to ease the development of outside systems in cooperation with smart contracts.

RECOMMENDATION

Add events for `setBlacklist()`, `setWhitelisted()`, `withdrawBNB()` and `withdrawERC20()` functions.

APPENDIX

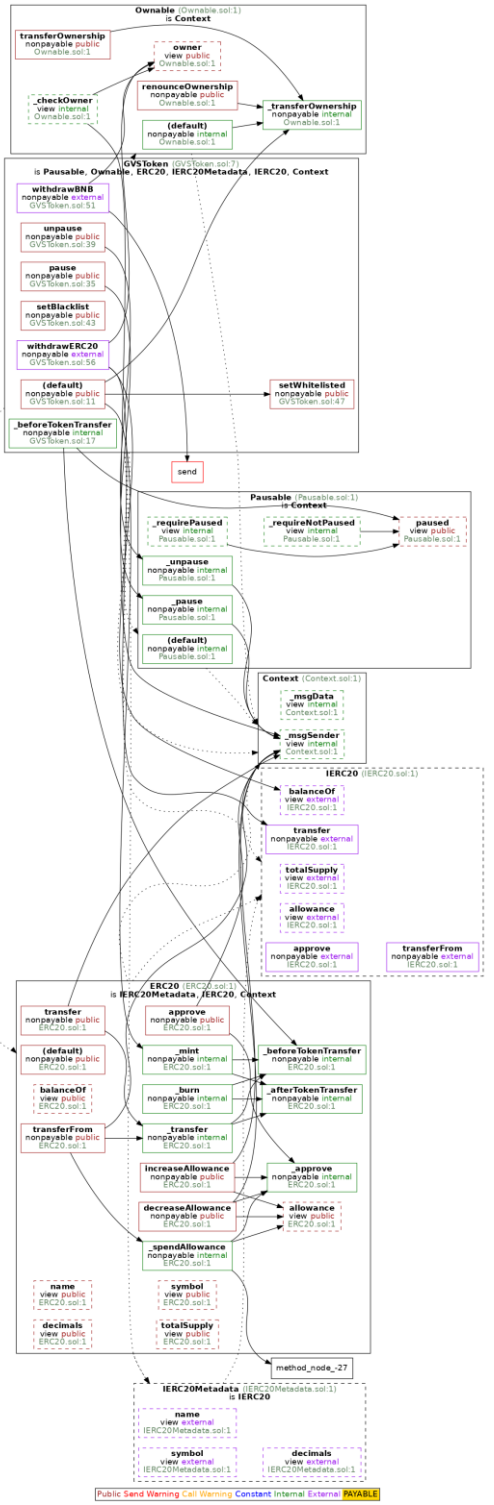

Image 1. GVSToken Smart Contract call graph

3. VERSION HISTORY

Version	Date	Status/Change	Created by
1.0	*Sep 6, 2023*	Public Report	Verichains Lab

Table 4. Report versions history