Submitted audit for **GbtIssunance** on 23 November 2023
Audit result: **Passed**

Token Address: - 0x19bfAF9843357aCb87dB558118AC9bC73662D98F
Name: GoldenBambooToken
Symbol: GBT
Decimals: 18
Network: Binance smart chain
Token Type: ERC20
Owner:- 0x0000000000000000000000000000000000000000
Deployer:- 0x0000000000000000000000000000000000000000
Token Supply: 100000000000000000000
Checksum: 27af7fb5e4ae1c007d623902f5e5a456
Testnet version:
The tests were performed using the contract deployed on the Binance smart chain Testnet, which can be found at the following address:
https://testnet.bscscan.com/address/0x19bfaf9843357acb87db558118ac9bc73662d98f#code

Tools:

1. Manual Review: The code has undergone a line-by-line review by the **Ace** team.
2. BSC Test Network: All tests were conducted on the BSC Test network, and each test has a corresponding transaction attached to it. These tests can be found in the "Functional Tests" section of the report.
3. Slither: The code has undergone static analysis using Slither.

Static Analysis

A static analysis of the code was performed using Slither.

Findings:
Critical: 0
High: 0
Medium: 0
Low: 2
Suggestions & Optimizations: 2

Centralization – Missing Zero Address

Severity: Low
function: setAddress
Status: Open
Overview:

functions can take a zero address as a parameter (0x00000...). If a function parameter of address type is not properly validated by checking for zero addresses, there could be serious consequences for the contract's functionality.

```
function setAddress(address param, uint status) external onlyOwner {
        if (status == 0) {
            _token = param;
        } else if (status == 1) {
            _broker = param;
        }
    }
```

Suggestion:
It is suggested that the address should not be zero or dead.

Optimization

Severity: Low
subject: Missing Events
Status: Open
Overview:
They serve as a mechanism for emitting and recording data onto the blockchain, making it transparent and easily accessible.

```
function setAddress(address param, uint status) external onlyOwner {
        if (status == 0) {
            _token = param;
        } else if (status == 1) {
            _broker = param;
        }
    }
```

Optimization

Severity: Informational
subject: floating Pragma Solidity version
Status: Open
Overview:
It is considered best practice to pick one compiler version and stick with it. With a floating pragma, contracts may accidentally be deployed using an outdated.

```
pragma solidity ^0.8.19;
```

Suggestion

Adding the latest constant version of solidity is recommended, as this prevents the unintentional deployment of a contract with an outdated compiler that contains unresolved bugs.

Optimization

Severity: Informational
subject: Remove safe math
Status: Open
Line: 98 - 133
Overview:
Compiler version above 0.8.0 has the ability to control arithmetic overflow/underflow, It is recommended to remove the unwanted code in order to avoid high gas fees.