Revoluzion Audit

Audit Report

Name	: **Gorileth Token**
Symbol	: **Goken**
Decimals	: **9**
Address	: **0x8A87f0122906Acd8Ebf1b7f45f350e27630a5036**
Owner	: **0x744b3edB7FfdD969e9e1e90B96Ac597107c4a03d**
Network	: **Binance Smart Chain (Mainnet)**
Type	: **BEP20**
Audited on	: **15 September 2022**
Updated on	: **21 September 2022**

Revoluzion Audit

Contents

Project Overview

Name	Gorileth Token
Symbol	Goken
Decimals	9
Total Supply	1,000,000,000
Tax	Buy 10% | Sell 15% — (Fixed Tax)
Compiler Version	v0.8.17+commit.8df45f5f
Optimization	Yes with 200 runs
License Type	MIT
Explorer Link	https://bscscan.com/address/0x8A87f0122906Acd8Ebf1b7f45f350e27630a5036
Create Tx	0xdf75f580e0700c3922f2b5eb74c2ae067d25210d1d811ac19fa3694d48d29d50
Creator	0x744b3edB7FfdD969e9e1e90B96Ac597107c4a03d
Featured Wallet	Marketing Wallet — 0x0A94282f2229d312F3924F78205cA1F26bC571Cb Team Wallet — 0x0C47C71772aca6762B2675519346e81414a2ebC8 Treasury Wallet — 0x332fDda1Ab3bDf1bb56659CC4e2A987C270A28dc
GitHub Link	https://github.com/gorileth/GorilethToken
Website	https://gorileth.com

Project Description

According to their website

Gorileth is a blockchain game centered around breedable and collectible digital gorillas. In this game, you can breed your gorillas with unique features and use them as to fight with your peers. While breeding your crabs, you could create a unique breed that could be valuable as a collectible. While the game centered around the NFT, the ecosystem itself will be using Gorileth Token (Goken) as a utility token for most of the transaction within the ecosystem.

Release Date : 24 September 2022

Category : Utility Token

Online Presence

About Website

Registrar : https://www.godaddy.com

Domain Expiration : 2023-10-03

SSL Certificate : Issued by Let's Encrypt

Official Links

Website	https://gorileth.com
Facebook	https://facebook.com/gorileth
Instagram	https://instagram.com/gorileth
Twitter	https://twitter.com/gorileth
LinkedIn	https://linkedin.com/company/gorileth
GitHub	https://github.com/gorileth
TikTok	https://tiktok.com/@gorileth
Reddit	https://reddit.com/user/gorileth
Medium	https://medium.com/@gorileth
Telegram Channel	https://t.me/ gorilethOC
Telegram Group	https://t.me/gorilethOG

The Team

About	The team has privately doxxed to Revoluzion by completing the tasks as listed below.
KYC Issuer	Revoluzion
Member's KYC'd	1
KYC Date	15 September 2022
Certificate Link	**https://github.com/RevoluzionToken/Revoluzion-Audits/tree/main/GorilethToken/certificate/KYC.png**
Task Completed	ID Verification — **Completed** Owner's wallet verification — **Completed**

Audit Overview

Threat Level

When conducting audit on smart contract(s), we first look for known vulnerabilities and issues within the code because any exploitation on such vulnerabilities and issues by malicious actors could potentially result in serious financial damage to the projects. All the issues and vulnerabilities will be categorized into the categories as provided below.

Critical

This category provides issues and vulnerabilities that are critical to the performance/functionality of the smart contract and should be fixed by project creator before moving to a live environment.

Medium

This category provides issues and vulnerabilities that are not that critical to the performance/functionality of the smart contract but is recommended to be fixed by project creator before moving to a live environment.

Minor

This category provides issues and vulnerabilities that are minor to the performance/functionality of the smart contract and can remain unfixed by project creator before moving to a live environment.

Informational

This category provides issues and vulnerability that have insignificant effect on the performance/functionality of the smart contract and can remain unfixed by project creator before moving to a live environment. However, fixing them can further improve the efficacy or security for features with a risk-free factor.

Notable Information

- Contract Owner cannot stop or pause transactions.
- Contract Owner cannot transfer tokens from specific address.
- Contract Owner cannot increase the distribution of liquidity taken more than 5%.
- Contract Owner cannot mint new tokens after deploying smart contract.
- Contract Owner cannot burn tokens from specific wallet.
- Contract Owner cannot blacklist wallets from selling.
- Fixed buy and sell fees hardcoded as 10% and 15% respectively.
- There are no compiler warnings when compiling the smart contracts.
- Contract is using safe Zeppelin modules.

Contract Diagnostic

Link for initial smart contract commit being audited on GitHub:

https://github.com/gorileth/GorilethToken/commit/fd2503a56f1d6c6ca8d97da543c44b4975525a15

CODE	SEVERITY	DESCRIPTION
SWC-103	Minor	A floating pragma is set.
SWC-120	Minor	Potential use of "block.number" as source of randomness.
SS	Informational	Function state shadowing other function.
US	Informational	State variable not initialized.
UR	Informational	Unused return value(s).
BE	Informational	Comparing to constant Boolean value.
DC	Informational	Dead code.
NC	Informational	Naming convention.
SN	Informational	Similar name.
CS	Informational	State variable that can be declared as constant.
EF	Informational	Public function can be declared as external.

SWC-103 — A floating pragma is set

SEVERITY	Minor
LOCATION(S)	GorilethToken.sol#L5
DESCRIPTION	The current pragma Solidity directive is set as ""^0.8.17"".
RECOMMENDATIONS	Project creator is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. It is important if the project rely on bytecode-level verification of the code.
STATUS	**FIXED by project creator** https://github.com/gorileth/GorilethToken/commit/eae6d5780038eb819d5a8279ff3cf86ed09443ae

SWC-120 — Potential use of "block.number" as source of randomness

SEVERITY	Minor
LOCATION(S)	GorilethToken.sol#1030, 1132, 1215
DESCRIPTION	The environment variable "block.number" looks like it might be used as a source of randomness to trigger a function.
RECOMMENDATIONS	We would recommend project owner to not use any of the environment variables like coinbase, gaslimit, block number and timestamp as sources of randomness since they are predictable and be aware that such usage could introduces a certain level of trust into miners. Keep in mind that malicious miner can manipulate the value of those variables and that any attackers could also predetermine the hashes of earlier blocks. However, based on our analysis, there's nothing to be done by project owner since in each of the "block.number" value was used as a means to keep track of time/epoch that relates to the trigger of a specific function.
STATUS	**N/A**

SS — Function state shadowing other function

SEVERITY	Informational — Medium
LOCATION(S)	GorilethToken.sol#L521, 523, 926, 927
DESCRIPTION	[GorilethToken._balances] (#L926) shadows [ERC20._balances] (#L521) [GorilethToken._allowances] (#L927) shadows [ERC20._allowances] (#L523)
RECOMMENDATIONS	Project creator can rename _balances and _allowances under GorilethToken.sol to something else or completely remove them.
STATUS	**FIXED by project creator** https://github.com/gorileth/GorilethToken/commit/eae6d5780038eb819d5a8279ff3cf86ed09443ae

US — State variable not initialized

SEVERITY	Informational — High
LOCATION(S)	GorilethToken.sol#L896
DESCRIPTION	[GorilethToken.xTeam] (#L896) is never initialized despite being used within [GorilethToken.swapBack()] (#L1177-1211)
RECOMMENDATIONS	Project creator needs to initialize xTeam variable under GorilethToken.sol. We recommend doing so on #L1021 since the typing error caused here is the reason that leads to this issue.
STATUS	**FIXED by project creator** https://github.com/gorileth/GorilethToken/commit/eae6d5780038eb819d5a8279ff3cf86ed09443ae

UR — Unused return value(s)

SEVERITY	Informational — Minor
LOCATION(S)	GorilethToken.sol#L1177-1211
DESCRIPTION	[GorilethToken.swapBack()] (#L1177-1211) ignores the return value at [GorilethToken.swapBack()] (#L1203-1205)
RECOMMENDATIONS	Based on our analysis, project creator doesn't need to do anything for this issue since it will be redundant.
STATUS	**N/A**

BE — Comparing to constant Boolean value

SEVERITY	Informational — Minor
LOCATION(S)	GorilethToken.sol#L943
DESCRIPTION	[GorilethToken.onlyBuybacker()] (#L940-943) compares to a Boolean constant at #L941
RECOMMENDATIONS	Project creator could optimize this line of code by directly referring to the state value of the mapping instead of redundantly comparing it to the constant Boolean value.
STATUS	**FIXED by project creator** https://github.com/gorileth/GorilethToken/commit/eae6d5780038eb819d5a8279ff3cf86ed09443ae

DC — Dead code

SEVERITY	Informational — Medium
LOCATION(S)	GorilethToken.sol#L25-27, 29-31, 713-735, 772-789, 824-836
DESCRIPTION	[Context._msgData()] (#L2527) is never used and should be removed. [Context._msgValue()] (#L29-31) is never used and should be removed. [ERC20._transfer()] (#L713-735) is never used and should be removed. [ERC20._burn()] (#L772-789) is never used and should be removed [ERC20._spendAllowance()] (#L824-836) is never used and should be removed
RECOMMENDATIONS	Based on our analysis, the ERC20 smart contract is the standard that is a direct fork from Open Zeppelin and were used within the contract itself. Although those functions were never used elsewhere, especially in GorilethToken, it's still okay to leave them be. However, for Context smart contract, it is okay for project creator to remove those functions to further optimize the smart contract since they are not used anywhere at all. Doing so will reduce the amount gas required when deploying the smart contract.
STATUS	**FIXED by project creator** https://github.com/gorileth/GorilethToken/commit/eae6d5780038eb819d5a8279ff3cf86ed09443ae

NC — Naming convention

SEVERITY	Informational — Minor
LOCATION(S)	GorilethToken.sol#L189, 1044, 1049
DESCRIPTION	[IUniswapV2Router.WETH()] (#L2527) is not in mixedCase. [GorilethToken.setSwapBackSettings()] (#L1044-1047) is not in mixedCase. [GorilethToken.setTargetLiquidity()] (#L1049-1052) is not in mixedCase.
RECOMMENDATIONS	Based on our analysis, the IUniswapV2Router smart contract is a direct fork from Uniswap. Although the name doesn't conform to the standard convention, it's still okay to leave it be to avoid from potentially breaking any external function. However, for GorilethToken smart contract, it is okay for project creator to update the name of the parameters in those functions so that they conform to the standard naming convention.
STATUS	**FIXED by project creator** https://github.com/gorileth/GorilethToken/commit/eae6d5780038eb819d5a8279ff3cf86ed09443ae

SN — Similar name

SEVERITY	Informational — Minor
LOCATION(S)	GorilethToken.sol#L194, 525, 948
DESCRIPTION	[IUniswapV2Router01.addLiquidity()] (#L194) has two parameters names that are too similar. [ERC20._totalSupply] (#L525) has name that is too similar to the totalSupply_ parameter in [GorilethToken.constructor()].
RECOMMENDATIONS	Based on our analysis, the IUniswapV2Router smart contract is a direct fork from Uniswap. Although their names are too similar, it's still okay to leave them be for the purpose of following the standard parameter declaration that is widely used as reference. However, for GorilethToken smart contract, it is okay for project creator to update the name while adhering to the standard naming convention.
STATUS	**FIXED by project creator** https://github.com/gorileth/GorilethToken/commit/eae6d5780038eb819d5a8279ff3cf86ed09443ae

CS — State variable that can be declared as constant

SEVERITY	Informational — Minor
LOCATION(S)	GorilethToken.sol#L896, 901, 902, 903
DESCRIPTION	[GorilethToken.xTeam] (#L896) should be constant. [GorilethToken.baseFee] (#L901) should be constant. [GorilethToken.totalFee] (#902) should be constant. [GorilethToken.feeDenominator] (#L903) should be constant.
RECOMMENDATIONS	Based on our analysis, these variables should be declared as constant since they don't change throughout smart contract. However, for xTeam, we believe it is not supposed to be a constant because the current constant state of this variable is resulted from typing error on #L1021 which is the reason that leads to this issue. After fixing that issue, xTeam should no longer remain as a constant value.
STATUS	**FIXED by project creator** https://github.com/gorileth/GorilethToken/commit/eae6d5780038eb819d5a8279ff3cf86ed09443ae

EF — Public function can be declared as external

SEVERITY	Informational — Medium
LOCATION(S)	GorilethToken.sol#L122-124, 130-133, 549-551, 557-559, 600-604, 645-654, 668-672, 688-697, 1070-1072, 1074-1081
DESCRIPTION	[Auth.renounceOwnership()] (#L122-124) should be declared as external. [Auth.transferOwnership()] (#L130-133) should be declared as external. [ERC20.name()] (#L549-551) should be declared as external. [ERC20.symbol()] (#L557-559) should be declared as external. [ERC20.transfer()] (#L600-604) should be declared as external together with [GorilethToken.transfer()] (#L1070-1072). [ERC20.transferFrom()] (#L645-654) should be declared as external together with [GorilethToken.transfer()] (#L1074-1081). [ERC20.increaseAllowance()] (#L668-672) should be declared as external. [ERC20.decreaseAllowance()] (#L688-697) should be declared as external.
RECOMMENDATIONS	Based on our analysis, it is best for project creator to change the visibility of these functions from public to external for the purpose of optimizing the smart contract since they are not used internally at all within any of the smart contract.
STATUS	**FIXED by project creator** https://github.com/gorileth/GorilethToken/commit/eae6d5780038eb819d5a8279ff3cf86ed09443ae

Changelog

21 September 2021 —
https://github.com/gorileth/GorilethToken/commit/0b1c16cd2391e559c25431f67db257a9f351a3c8

Due to the issue with verifying the Gorileth Token smart contract on BSCScan, the creator has made some modification on the smart contract. The changes are as follow:

- Completely remove ERC20 smart contract dependency.
- Replace Gorileth Token dependency on ERC20 to IERC20 and IERC20Metadata.
- Change visibility of router, DEAD and ZERO from private to public.
- Add in name, symbol, decimals and totalSupply variable to Gorileth Token smart contract to replace similar variable dependencies inherit from ERC20 smart contract.
- Replace hardcoded router address with variable that need to be set upon deployment.

Disclaimer

This report only shows findings based on our limited project analysis according to the good industry practice from the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, overall online presence and team transparency details of which are set out in this report. To get a full view of our analysis, **it is important for you to read the full report.** Under no circumstances did Revoluzion Audit receive a payment to manipulate those results or change the awarding badge that we will be adding in our website. **Our team provides no guarantees against the sale of team tokens or the removal of liquidity by the project** audited in this document.

While **we have done our best to conduct thorough analysis to produce this report**, it is crucial to note that you should not rely solely on this report and use the content provided in this document as financial advice or a reason to buy any investment. The Our team disclaims any liability against us for the resulting losses based on the you decision made by relying on the content of this report. **You must conduct your own independent investigations before making any decisions** to protect yourselves from being scammed. We go into more detail on this in the disclaimer clause in the next page — please make sure to read it in full.

Full Disclaimer Clause

By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy all copies of this report downloaded and/or printed by you. This report is provided for information purposes only, on a non-reliance basis and does not constitute to any investment advice.

No one shall have any right to rely on the report or its contents, and Revoluzion Audit and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers, and other representatives) (collectively known as Revoluzion) owe no duty of care towards you or any other person, nor do we make any warranty or representation to any person on the accuracy or completeness of the report.

The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and Revoluzion hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report.

Except and only to the extent that it is prohibited by law, Revoluzion hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Revoluzion, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report.

The analysis of the security is purely based on the smart contracts, website, social media, and team.