soken

SMART CONTRACT SECURITY AUDIT

FWC Legends

Scan and check this report was posted at Soken Github

October, 2022

Website: soken.io

Table of Contents

Disclaimer

This is a comprehensive report based on our automated and manual examination of cybersecurity vulnerabilities and framework flaws of the project's smart contract.

Reading the full analysis report is essential to build your understanding of project's security level. It is crucial to take note, though we have done our best to perform this analysis and report, that you should not rely on the our research and cannot claim what it states or how we created it.

Before making any judgments, you have to conduct your own independent research.

We will discuss this in more depth in the following disclaimer - please read it fully.

DISCLAIMER: You agree to the terms of this disclaimer by reading this report or any portion thereof. Please stop reading this report and remove and delete any copies of this report that you download and/or print if you do not agree to these conditions. Scan and verify report's presence in the GitHub repository by a qr-code at the title page. This report is for non-reliability information only and does not represent investment advice. No one shall be entitled to depend on the report or its contents, and Soken and its affiliates shall not be held responsible to you or anyone else, nor shall Soken provide any guarantee or representation to any person with regard to the accuracy or integrity of the report.

Without any terms, warranties or other conditions other than as set forth in that exclusion and Soken excludes hereby all representations, warrants, conditions and other terms (including, without limitation, guarantees implied by the law of satisfactory quality, fitness for purposes and the use of reasonable care and skills).

The report is provided as "as is" and does not contain any terms and conditions. Except as legally banned, Soken disclaims all responsibility and responsibilities and no claim against Soken is made to any amount or type of loss or damages (without limitation, direct, indirect, special, punitive, consequential or pure economic loses or losses) that may be caused by you or any other person, or any damages or damages, including without limitations (whether innocent or negligent).

Security analysis is based only on the smart contracts. No applications or operations were reviewed for security. No product code has been reviewed.

Procedure

Our analysis contains following steps:

1. Project Analysis;

2. Manual analysis of smart contracts:
   - Deploying smart contracts on any of the network(Ropsten/Rinkeby) using Remix IDE
   - Hashes of all transaction will be recorded
   - Behaviour of functions and gas consumption is noted, as well.

3. Unit Testing:
   - Smart contract functions will be unit tested on multiple parameters and under multiple conditions to ensure that all paths of functions are functioning as intended.
   - In this phase intended behaviour of smart contract is verified.
   - In this phase, we would also ensure that smart contract functions are not consuming unnecessary gas.
   - Gas limits of functions will be verified in this stage.

4. Automated Testing:
   - Mythril
   - Oyente
   - Manticore
   - Solgraph

Terminology

We categorize the finding into 4 categories based on their vulnerability:

- Low-severity issue — less important, must be analyzed
- Medium-severity issue — important, needs to be analyzed and fixed
- High-severity issue —important, might cause vulnerabilities, must be analyzed and fixed
- Critical-severity issue —serious bug causes, must be analyzed and fixed.

Limitations

The security audit of Smart Contract cannot cover all vulnerabilities. Even if no vulnerabilities are detected in the audit, there is no guarantee that future smart contracts are safe. Smart contracts are in most cases safeguarded against specific sorts of attacks. In order to find as many flaws as possible, we carried out a comprehensive smart contract audit. Audit is a document that is not legally binding and guarantees nothing.

Basic Security Recommendation

Unlike hardware and paper wallets, hot wallets are connected to the internet and store private keys online, which exposes them to greater risk. If a company or an individual holds significant amounts of cryptocurrency in a hot wallet, they should consider using MultiSig addresses. Wallet security is enhanced when private keys are stored in different locations and are not controlled by a single entity.
More info: **https://blog.soken.io/how-to-gnosis-multisig-1c6c0860586f**

Token Contract Details for 31.10.2022

Contract Name: **BEP20Token**

Deployed address: **0x83ADB07bB91dDDE95A24982F1B2D343963ba3995**

Total Supply: **1,000,000**

Token Tracker: **Fwcl**

Decimals: **9**

Token holders: **153**

Transactions count: **576**

Top 100 holders dominance: **99.99%**

Audit Details

Project Name: **FWC Legends**

Language: **Solidity**

Compiler Version: **v0.5.16**

Blockchain: **BSC**

Social Profiles

Project Website: **https://fwclegends.games/**

Project Twitter: **https://twitter.com/fwclegends**

Project Telegram: **https://t.me/Fwclegendsofficial**

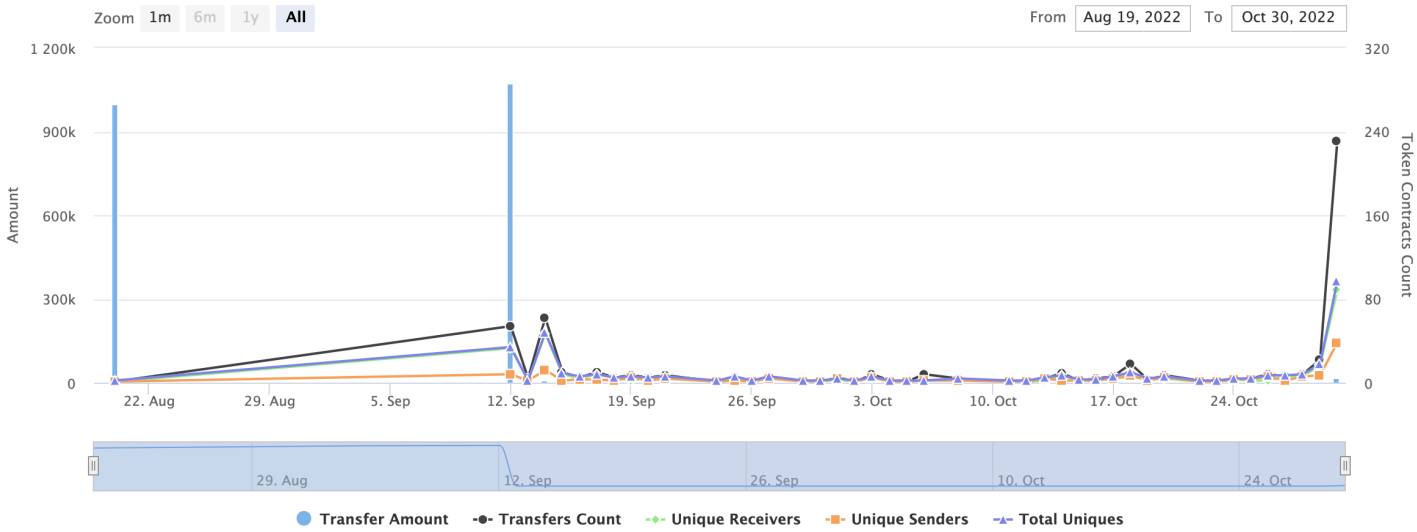
Project Reddit: **https://www.reddit.com/user/fwclegends**

Project Medium: **https://medium.com/@Legends.Game/about**

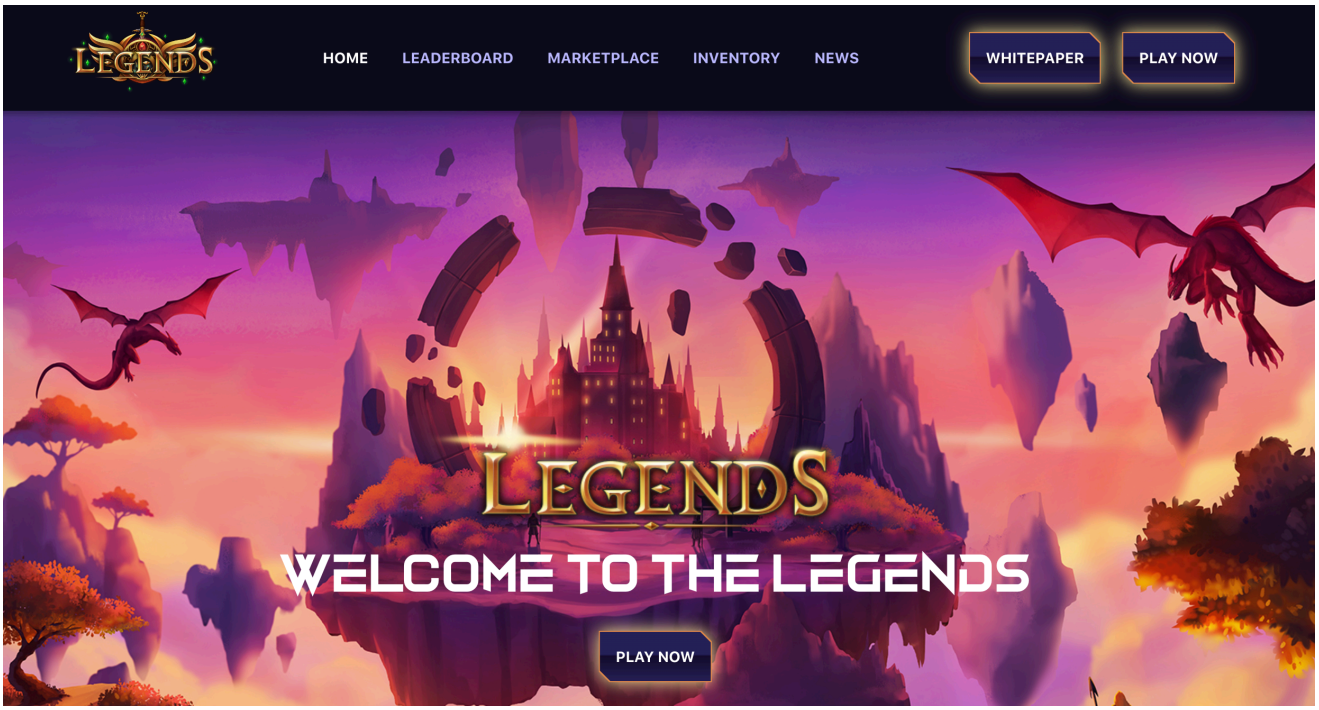
Project Instagram: **https://www.instagram.com/fwclegends/**

Project Github: **https://github.com/Fwclegends**

Token Analytics



Project Website Overview



✓JavaScript errors hasn't been found.
✓Malware pop-up windows hasn't been detected.
✓No issues with loading elements, code, or stylesheets.

Project Website SSL Certification

Issued To

Common Name (CN)	fwclegends.games
Organization (O)	<Not Part Of Certificate>
Organizational Unit (OU)	<Not Part Of Certificate>

Issued By

Common Name (CN)	R3
Organization (O)	Let's Encrypt
Organizational Unit (OU)	<Not Part Of Certificate>

Vulnerabilities checking

Issue Description	Checking Status
Compiler Errors	Completed
Delays in Data Delivery	Completed
Re-entrancy	Completed
Transaction-Ordering Dependence	Completed
Timestamp Dependence	Completed
Shadowing State Variables	Completed
DoS with Failed Call	Completed
DoS with Block Gas Limit	Completed
Outdated Complier Version	Completed
Assert Violation	Completed
Use of Deprecated Solidity Functions	Completed
Integer Overflow and Underflow	Completed
Function Default Visibility	Completed
Malicious Event Log	Completed
Math Accuracy	Completed
Design Logic	Completed
Fallback Function Security	Completed
Cross-function Race Conditions	Completed
Safe Zeppelin Module	Completed

Security Issues

1) Incorrect Access Control: High-severity. L440-443, L459-462, L476-L480, L494-497, L513-516

Access control plays an important role in segregation of privileges in smart contracts and other applications. If this is misconfigured or not properly validated on sensitive functions, it may lead to loss of funds, tokens and in some cases compromise of the smart contract. The contract BEP20Token is importing an access control library @openzeppelin/contracts/access/Ownable.sol but the function decreaseAllowance is missing the modifier onlyOwner.

2) Presence of Overpowered Role: Informational. L345-348, L354-357.

The overpowered owner (i.e., the person who has too much power) is a project design where the contract is tightly coupled to their owner (or owners); only they can manually invoke critical functions. Due to the fact that this function is only accessible from a single address, the system is heavily dependent on the address of the owner. In this case, there are scenarios that may lead to undesirable consequences for investors, e.g., if the private key of this address is compromised, then an attacker can take control of the contract.

Conclusion for project owner

High and informational-severity issues exist within smart contracts.

NOTE: Please check the disclaimer above and note, that audit makes no statements or warranties on business model, investment attractiveness or code sustainability. Contract security report for community

SECURITY REPORT
FOR COMMUNITY

FWC Legends

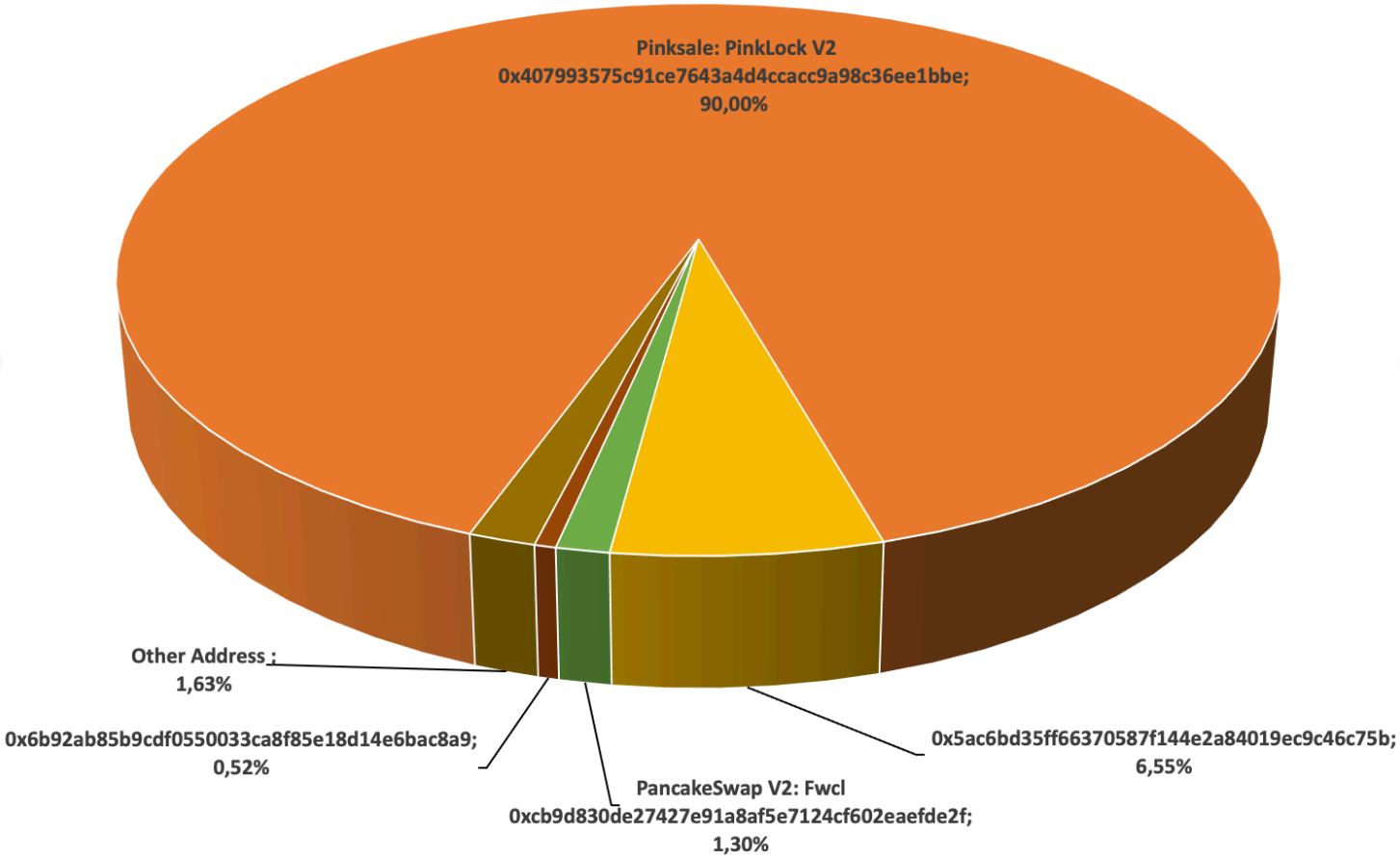
soken

Whitepaper of the project

The whitepaper of FWC Legends project has been verified on behalf of Soken team.

Whitepaper link: **https://legends-game-whitepaper.gitbook.io/legends-game/**

Fwcl Token Distribution



Fwcl Top 10 Holders

Rank	Address	Quantity (Token)	Percentage
1	Pinksale: PinkLock V2	900,000	90.0000%
2	0x5ac6bd35ff66370587f144e2a84019ec9c46c75b	65,480.688174507	6.5481%
3	PancakeSwap V2: Fwcl	12,993.919796838	1.2994%
4	0x6b92ab85b9cdf0550033ca8f85e18d14e6bac8a9	5,222.541447097	0.5223%
5	0xb8ba0729807601182e94e453384799189d37d2e5	887.592258438	0.0888%
6	0x23f99de24fbc3f441a126d3ddfa4b8830b2d998f	800.221015853	0.0800%
7	0x2218ae1d93e935beaebad62e54eb1ddc822858c1	780.991533458	0.0781%
8	0x5a2afdeff3c7c7e7cc20789bb4ec07c3cb5ad87b	709.209352333	0.0709%
9	0xa13b6fde0a47f3d199bc60742daf209f85576fe3	701.531685194	0.0702%
10	0x3601446fbede36fe840947e3d96fea768921f149	589.310166284	0.0589%

Soken Contact Info

Website: www.soken.io

Mob: (+1)416-875-4174

32 Britain Street, Toronto, Ontario, Canada

Telegram: @team_soken

GitHub: sokenteam

Twitter: @soken_team