TechRate

Blockchain solutions and consulting

Smart Contract Security Audit

Audit details:

Audited project:	Darkwing Finance
Deployer address	0xf3c31dad4e9d4a4ad8d4d19d8f9619cb55a6a886
Blockchain:	Binance Smart Chain
Project website:	https://darkwingfinance.com

April, 2021
TechRate

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn’t say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the below disclaimer below – please make sure to read it in full.

DISCLAIMER: 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 any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice. No one shall have any right to rely on the report or its contents, and TechRate and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (TechRate) owe no duty of care towards you or any other person, nor does TechRate 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 TechRate 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, TechRate hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against TechRate, 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 alone. No applications or operations were reviewed for security. No product code has been reviewed.

Background

TechRate was commissioned by Darkwing Finance to perform an audit of smart contracts:

- *https://bscscan.com/address/0x203139aA1e727a58838f0bE59440AfFbC746f78A#code*
- *https://bscscan.com/address/0x98369d5e8fDEc381e340d9a835898cA8Bf5ADdE6#code*
- *https://bscscan.com/address/0x887f6946DC46095c66fE48f93bA3aDe6ea1b7f22#code*
- *https://bscscan.com/address/0xC284C1e90efc4e5B4a343408aA603B5a85417c4A#code*

The purpose of the audit was to achieve the following:

- Ensure that the smart contract functions as intended.
- Identify potential security issues with the smart contract.

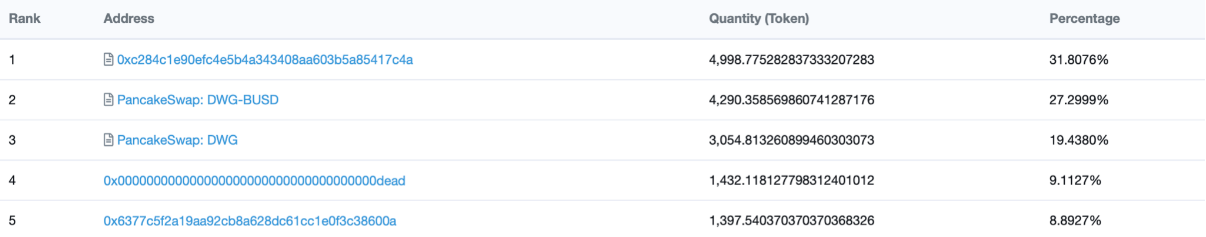
The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Contracts details

Token contract details for 11.04.2021.

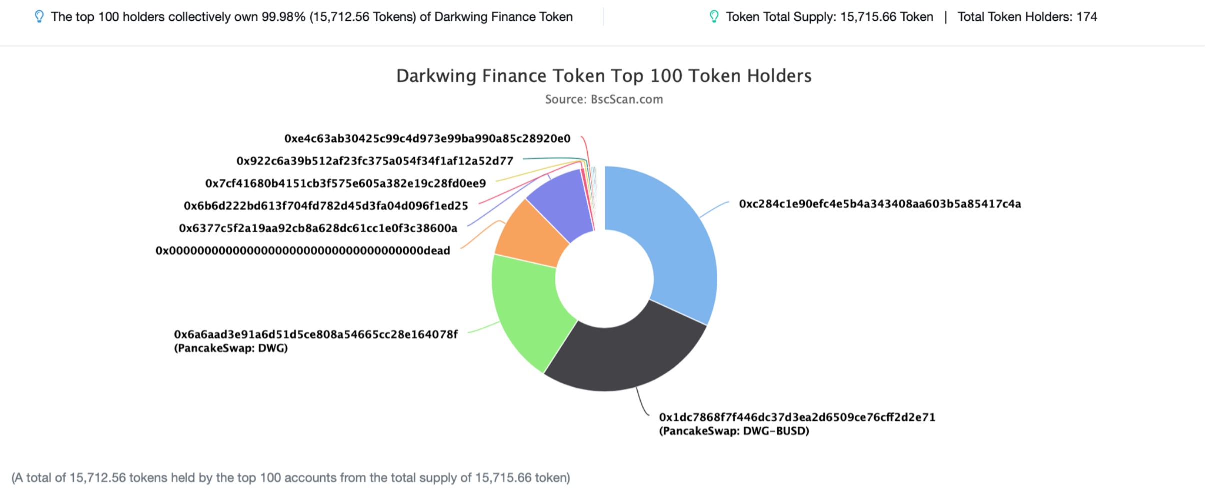
Contract name:	Darkwing Finance Token
Compiler version:	v0.6.12+commit.27d51765
Contract address:	0x98369d5e8fDEc381e340d9a835898cA8Bf5ADdE6
Total supply:	15_708_237_261_503_794_740_637
Token ticker:	DWG
Decimals:	18
Token holders:	174
Transactions count:	23638
Top 100 holders dominance:	99 %
Contract deployer address:	0xf3c31dad4e9d4a4ad8d4d19d8f9619cb55a6a886
Contract's current owner address:	0xc284c1e90efc4e5b4a343408aa603b5a85417c4a

Darkwing Finance top 5 token holders



Rank	Address	Quantity (Token)	Percentage
1	0xc284c1e90efc4e5b4a343408aa603b5a85417c4a	4,998.775282837333207283	31.8076%
2	PancakeSwap: DWG-BUSD	4,290.358569860741287176	27.2999%
3	PancakeSwap: DWG	3,054.813260899460303073	19.4380%
4	0x000000000000000000000000000000000000dead	1,432.118127798312401012	9.1127%
5	0x6377c5f2a19aa92cb8a628dc61cc1e0f3c38600a	1,397.540370370370368326	8.8927%

Darkwing Finance top 100 token distribution



The top 100 holders collectively own 99.98% (15,712.56 Tokens) of Darkwing Finance Token

Token Total Supply: 15,715.66 Token | Total Token Holders: 174

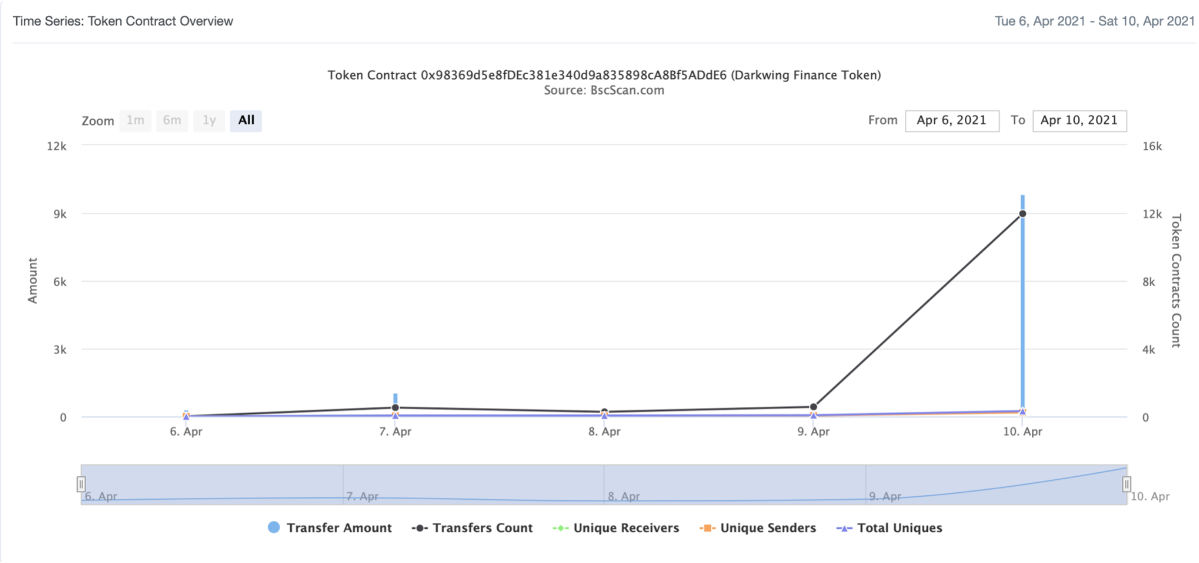
Darkwing Finance Token Top 100 Token Holders

Source: BscScan.com

0xe4c63ab30425c99c4d973e99ba990a85c28920e0
0x922c6a39b512af23fc375a054f34f1af12a52d77
0x7cf41680b4151cb3f575e605a382e19c28fd0ee9
0x6b6d222bd613f704fd782d45d3fa04d096f1ed25
0x6377c5f2a19aa92cb8a628dc61cc1e0f3c38600a
0x000000000000000000000000000000000000dead
0xc284c1e90efc4e5b4a343408aa603b5a85417c4a
0x6a6aad3e91a6d51d5ce808a54665cc28e164078f (PancakeSwap: DWG)
0x1dc7868f7f446dc37d3ea2d6509ce76cff2d2e71 (PancakeSwap: DWG-BUSD)

(A total of 15,712.56 tokens held by the top 100 accounts from the total supply of 15,715.66 token)

Darkwing Finance contract interaction details



Time Series: Token Contract Overview

Tue 6, Apr 2021 - Sat 10, Apr 2021

Token Contract 0x98369d5e8fDEc381e340d9a835898cA8Bf5ADdE6 (Darkwing Finance Token)

Source: BscScan.com

Zoom 1m 6m 1y **All**

From Apr 6, 2021 To Apr 10, 2021

Masterchef contract details for 11.04.2021.

Contract name:	MasterChef
Compiler version:	v0.6.12+commit.27d51765
Contract address:	0xC284C1e90efc4e5B4a343408aA603B5a85417c4A
Dev address:	0x6377c5f2a19aa92cb8a628dc61cc1e0f3c38600a
Fee address:	0xb82241e90ad25bb39e54a7e8214b1d9c7c560e14
DWG contract address:	0x98369d5e8fdec381e340d9a835898ca8bf5adde6
DWG per block:	1_000_000_000_000_000_000
Contract owner address:	0x203139aa1e727a58838f0be59440affbc746f78a
Pool length:	29
Start block:	6455000
Total alloc point:	13500
Bonus multiplier:	1
Max deposit fee:	100 %
Referral commission rate:	200
DWG referral address:	0x887f6946dc46095c66fe48f93ba3ade6ea1b7f22
Max referral commission rate:	2000
Emission reduction period blocks:	9600

MasterChef contract Pools info:

Pool with id 0:

lpToken *address* : 0x1dC7868f7f446dC37D3Ea2D6509cE76CFF2d2e71
allocPoint *uint256* : 4000
lastRewardBlock *uint256* : 6469293
accDwgPerShare *uint256* : 2312562116232
depositFeeBP *uint16* : 0

Pool with id 1:

lpToken *address* : 0x6a6AaD3e91a6d51D5CE808A54665CC28E164078F
allocPoint *uint256* : 2400
lastRewardBlock *uint256* : 6469398
accDwgPerShare *uint256* : 52061679569691
depositFeeBP *uint16* : 0

Pool with id 2:

lpToken *address* : 0x1B96B92314C44b159149f7E0303511fB2Fc4774f
allocPoint *uint256* : 500
lastRewardBlock *uint256* : 6469321
accDwgPerShare *uint256* : 1718082603892
depositFeeBP *uint16* : 400

Pool with id 3:

lpToken *address* : 0xc15fa3E22c912A276550F3E5FE3b0Deb87B55aCd
allocPoint *uint256* : 400
lastRewardBlock *uint256* : 6468898
accDwgPerShare *uint256* : 64638873533
depositFeeBP *uint16* : 400

Pool with id 4:

lpToken *address* : 0x7561EEe90e24F3b348E1087A005F78B4c8453524
allocPoint *uint256* : 600
lastRewardBlock *uint256* : 6468965
accDwgPerShare *uint256* : 339670247200022
depositFeeBP *uint16* : 400

Pool with id 5:

lpToken *address* : 0x70D8929d04b60Af4fb9B58713eBcf18765aDE422
allocPoint *uint256* : 600
lastRewardBlock *uint256* : 6468966
accDwgPerShare *uint256* : 50706755139678
depositFeeBP *uint16* : 400

Pool with id 6:

lpToken *address* : 0x3aB77e40340AB084c3e23Be8e5A6f7afed9D41DC
allocPoint *uint256* : 400
lastRewardBlock *uint256* : 6469416
accDwgPerShare *uint256* : 93940433446
depositFeeBP *uint16* : 400

Pool with id 7:

lpToken *address* : 0x680Dd100E4b394Bda26A59dD5c119A391e747d18
allocPoint *uint256* : 400
lastRewardBlock *uint256* : 6469254
accDwgPerShare *uint256* : 97621756534
depositFeeBP *uint16* : 400

Pool with id 8:

lpToken *address* : 0xbCD62661A6b1DEd703585d3aF7d7649Ef4dcDB5c
allocPoint *uint256* : 600
lastRewardBlock *uint256* : 6468967
accDwgPerShare *uint256* : 15086622231173
depositFeeBP *uint16* : 400

Pool with id 9:

lpToken *address* : 0x0Ed8E0A2D99643e1e65CCA22Ed4424090B8B7458
allocPoint *uint256* : 200
lastRewardBlock *uint256* : 6469383
accDwgPerShare *uint256* : 485861126637
depositFeeBP *uint16* : 400

Pool with id 10:

lpToken *address* : 0xA527a61703D82139F8a06Bc30097cC9CAA2df5A6
allocPoint *uint256* : 200
lastRewardBlock *uint256* : 6468558
accDwgPerShare *uint256* : 4330042487444
depositFeeBP *uint16* : 400

Pool with id 11:

lpToken *address* : 0x98369d5e8fDEc381e340d9a835898cA8Bf5ADdE6
allocPoint *uint256* : 1000
lastRewardBlock *uint256* : 6469390
accDwgPerShare *uint256* : 1083876098612
depositFeeBP *uint16* : 0

Pool with id 12:

lpToken *address* : 0xe9e7CEA3DedcA5984780Bafc599bD69ADd087D56
allocPoint *uint256* : 200
lastRewardBlock *uint256* : 6469141
accDwgPerShare *uint256* : 39738693906
depositFeeBP *uint16* : 400

Pool with id 13:

lpToken *address* : 0xbb4CdB9CBd36B01bD1cBaEBF2De08d9173bc095c
allocPoint *uint256* : 300
lastRewardBlock *uint256* : 6469471
accDwgPerShare *uint256* : 20552231522030
depositFeeBP *uint16* : 400

Pool with id 14:

lpToken *address* : 0x55d398326f99059fF775485246999027B3197955
allocPoint *uint256* : 100
lastRewardBlock *uint256* : 6469143
accDwgPerShare *uint256* : 42727775579
depositFeeBP *uint16* : 400

Pool with id 15:

lpToken *address* : 0x7130d2A12B9BCbFAe4f2634d864A1Ee1Ce3Ead9c
allocPoint *uint256* : 200
lastRewardBlock *uint256* : 6469515
accDwgPerShare *uint256* : 4089772491417756
depositFeeBP *uint16* : 400

Pool with id 16:

lpToken *address* : 0x2170Ed0880ac9A755fd29B2688956BD959F933F8
allocPoint *uint256* : 200
lastRewardBlock *uint256* : 6469521
accDwgPerShare *uint256* : 148465952122995
depositFeeBP *uint16* : 400

Pool with id 17:

lpToken *address* : 0x1AF3F329e8BE154074D8769D1FFa4eE058B1DBc3
allocPoint *uint256* : 100
lastRewardBlock *uint256* : 6469512
accDwgPerShare *uint256* : 49035825627
depositFeeBP *uint16* : 400

Pool with id 18:

```
lpToken   address :  0x8AC76a51cc950d9822D68b83fE1Ad97B32Cd580d
allocPoint   uint256 :  100
lastRewardBlock   uint256 :  6469531
accDwgPerShare   uint256 :  65459967309
depositFeeBP   uint16 :  400
```

Pool with id 19:

```
lpToken   address :  0x7083609fCE4d1d8Dc0C979AAb8c869Ea2C873402
allocPoint   uint256 :  200
lastRewardBlock   uint256 :  6469298
accDwgPerShare   uint256 :  1912729579125
depositFeeBP   uint16 :  400
```

Pool with id 20:

```
lpToken   address :  0x0E09FaBB73Bd3Ade0a17ECC321fD13a19e81cE82
allocPoint   uint256 :  100
lastRewardBlock   uint256 :  6469298
accDwgPerShare   uint256 :  1433049822830
depositFeeBP   uint16 :  400
```

Pool with id 21:

```
lpToken   address :  0x5Ac52EE5b2a633895292Ff6d8A89bB9190451587
allocPoint   uint256 :  100
lastRewardBlock   uint256 :  6469299
accDwgPerShare   uint256 :  951021808451
depositFeeBP   uint16 :  400
```

Pool with id 22:

```
lpToken   address :  0xa184088a740c695E156F91f5cC086a06bb78b827
allocPoint   uint256 :  100
lastRewardBlock   uint256 :  6468798
accDwgPerShare   uint256 :  105854534111704
depositFeeBP   uint16 :  400
```

Pool with id 23:

```
lpToken   address :  0xF952Fc3ca7325Cc27D15885d37117676d25BfdA6
allocPoint   uint256 :  100
lastRewardBlock   uint256 :  6468798
accDwgPerShare   uint256 :  689126701734
depositFeeBP   uint16 :  400
```

Pool with id 24:

lpToken *address* : 0x8148b58393f00b4B379cBEb8018d3445E0b636a0
allocPoint *uint256* : 100
lastRewardBlock *uint256* : 6469411
accDwgPerShare *uint256* : 9009738219
depositFeeBP *uint16* : 400

Pool with id 25:

lpToken *address* : 0x57067A6BD75c0E95a6A5f158455926e43E79BeB0
allocPoint *uint256* : 100
lastRewardBlock *uint256* : 6469038
accDwgPerShare *uint256* : 5424074107333
depositFeeBP *uint16* : 400

Pool with id 26:

lpToken *address* : 0xCa3F508B8e4Dd382eE878A314789373D80A5190A
allocPoint *uint256* : 100
lastRewardBlock *uint256* : 6469591
accDwgPerShare *uint256* : 306057161366435
depositFeeBP *uint16* : 400

Pool with id 27:

lpToken *address* : 0x7A9f28EB62C791422Aa23CeAE1dA9C847cBeC9b0
allocPoint *uint256* : 50
lastRewardBlock *uint256* : 6469622
accDwgPerShare *uint256* : 164595425209
depositFeeBP *uint16* : 400

Pool with id 28:

lpToken *address* : 0x5eF5994fA33FF4eB6c82d51ee1DC145c546065Bd
allocPoint *uint256* : 50
lastRewardBlock *uint256* : 6469176
accDwgPerShare *uint256* : 205788900683
depositFeeBP *uint16* : 400

Issues Checking Status

№	Issue description.	Checking status
1	Compiler errors.	Passed
2	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Front running.	Passed
6	Timestamp dependence.	Passed
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Passed
10	Methods execution permissions.	Passed
11	Economy model of the contract.	Passed
12	The impact of the exchange rate on the logic.	Passed
13	Private user data leaks.	Passed
14	Malicious Event log.	Passed
15	Scoping and Declarations.	Passed
16	Uninitialized storage pointers.	Passed
17	Arithmetic accuracy.	Passed
18	Design Logic.	Some issues
19	Cross-function race conditions.	Passed
20	Safe Open Zeppelin contracts implementation and usage.	Passed
21	Fallback function security.	Passed

Security Issues

High Severity Issues

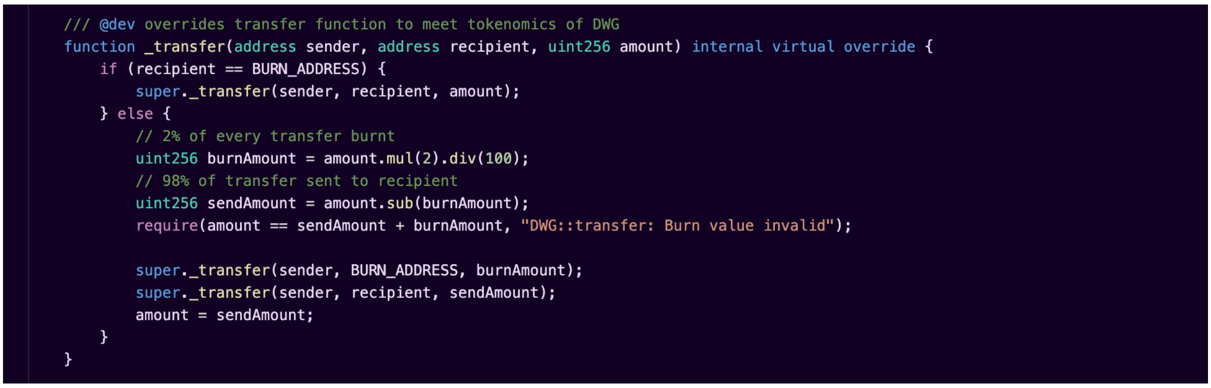
No high severity issues found.

Medium Severity Issues

1. Wrong burning

Issue:

There is sending burnable tokens to the dead address in overridden function _transfer in Darkwing token contract.



```
/// @dev overrides transfer function to meet tokenomics of DWG
function _transfer(address sender, address recipient, uint256 amount) internal virtual override {
    if (recipient == BURN_ADDRESS) {
        super._transfer(sender, recipient, amount);
    } else {
        // 2% of every transfer burnt
        uint256 burnAmount = amount.mul(2).div(100);
        // 98% of transfer sent to recipient
        uint256 sendAmount = amount.sub(burnAmount);
        require(amount == sendAmount + burnAmount, "DWG::transfer: Burn value invalid");

        super._transfer(sender, BURN_ADDRESS, burnAmount);
        super._transfer(sender, recipient, sendAmount);
        amount = sendAmount;
    }
}
```

Recommendation:

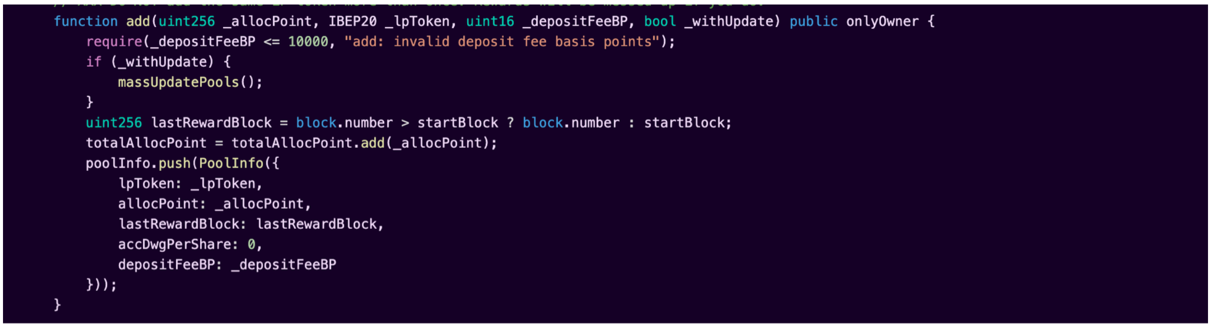
We recommend using the burn function for burning funds so the total supply will also decrease.

Low Severity Issues

1. add function issue

Issue:

If some LP token is added to the contract twice using function add, then the total amount of reward dwgReward in function updatePool will be incorrect.



```
function add(uint256 _allocPoint, IBEP20 _lpToken, uint16 _depositFeeBP, bool _withUpdate) public onlyOwner {
    require(_depositFeeBP <= 10000, "add: invalid deposit fee basis points");
    if (_withUpdate) {
        massUpdatePools();
    }
    uint256 lastRewardBlock = block.number > startBlock ? block.number : startBlock;
    totalAllocPoint = totalAllocPoint.add(_allocPoint);
    poolInfo.push(PoolInfo({
        lpToken: _lpToken,
        allocPoint: _allocPoint,
        lastRewardBlock: lastRewardBlock,
        accDwgPerShare: 0,
        depositFeeBP: _depositFeeBP
    }));
}
```

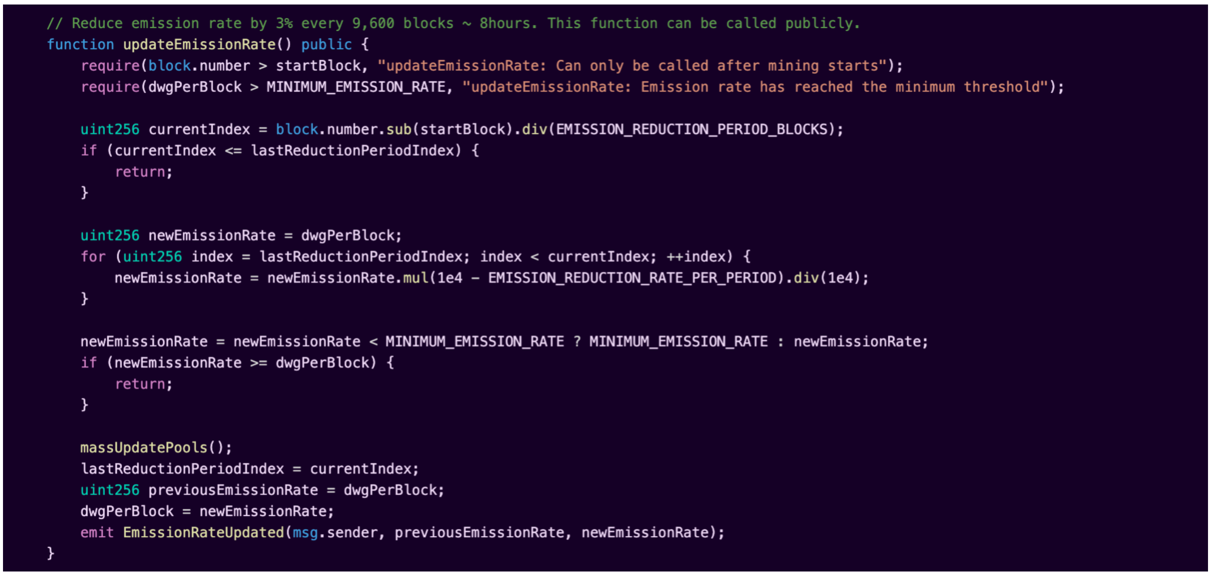
Recommendation:

Add the mapping from address to bool and check that same address will not be added twice.

2. Block gas limit

Issue:

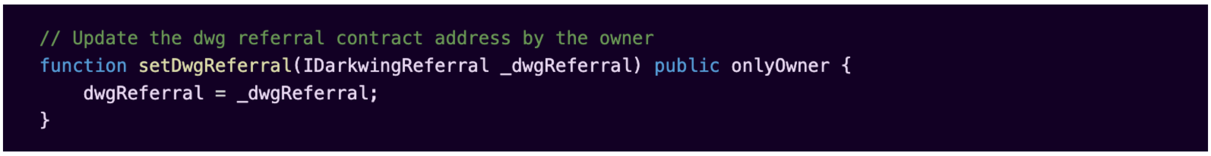
The updateEmissionRate function can fail due to block gas limit if the pool size is too big.



```
// Reduce emission rate by 3% every 9,600 blocks ~ 8hours. This function can be called publicly.
function updateEmissionRate() public {
    require(block.number > startBlock, "updateEmissionRate: Can only be called after mining starts");
    require(dwgPerBlock > MINIMUM_EMISSION_RATE, "updateEmissionRate: Emission rate has reached the minimum threshold");

    uint256 currentIndex = block.number.sub(startBlock).div(EMISSION_REDUCTION_PERIOD_BLOCKS);
    if (currentIndex <= lastReductionPeriodIndex) {
        return;
    }

    uint256 newEmissionRate = dwgPerBlock;
    for (uint256 index = lastReductionPeriodIndex; index < currentIndex; ++index) {
        newEmissionRate = newEmissionRate.mul(1e4 - EMISSION_REDUCTION_RATE_PER_PERIOD).div(1e4);
    }

    newEmissionRate = newEmissionRate < MINIMUM_EMISSION_RATE ? MINIMUM_EMISSION_RATE : newEmissionRate;
    if (newEmissionRate >= dwgPerBlock) {
        return;
    }

    massUpdatePools();
    lastReductionPeriodIndex = currentIndex;
    uint256 previousEmissionRate = dwgPerBlock;
    dwgPerBlock = newEmissionRate;
    emit EmissionRateUpdated(msg.sender, previousEmissionRate, newEmissionRate);
}
```

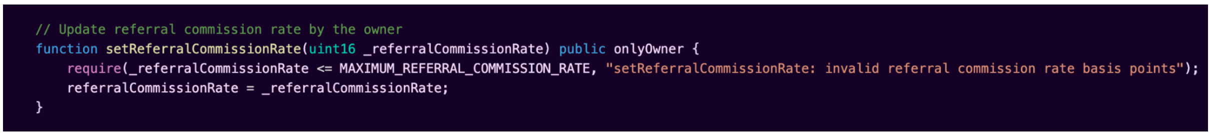
Owner privileges

❑ Owner can change the referral contract to a new not audited contract. (Ownership of Masterchef transferred to the Timelock contract now)



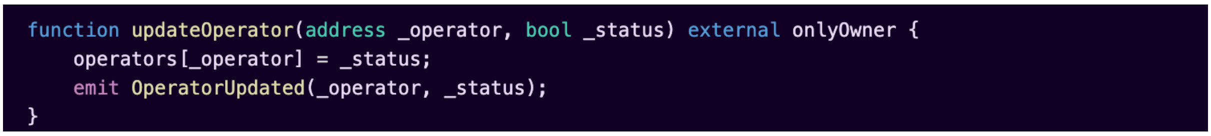
```
// Update the dwg referral contract address by the owner
function setDwgReferral(IDarkwingReferral _dwgReferral) public onlyOwner {
    dwgReferral = _dwgReferral;
}
```

❑ Owner can change the referral commission rate. (Ownership of Masterchef transferred to the Timelock contract now)



```
// Update referral commission rate by the owner
function setReferralCommissionRate(uint16 _referralCommissionRate) public onlyOwner {
    require(_referralCommissionRate <= MAXIMUM_REFERRAL_COMMISSION_RATE, "setReferralCommissionRate: invalid referral commission rate basis points");
    referralCommissionRate = _referralCommissionRate;
}
```

❑ Owner can add anyone as an operator of a referral contract.



```
function updateOperator(address _operator, bool _status) external onlyOwner {
    operators[_operator] = _status;
    emit OperatorUpdated(_operator, _status);
}
```

Conclusion

Smart contracts do not contain any high severity issues!

Techrate note:
Please check the disclaimer above and note, the audit makes no statements or warranties on business model, investment attractiveness or code sustainability. The report is provided for the only contract mentioned in the report and does not include any other potential contracts deployed by Owner.