Techrate1 Techrate

TechRate

AUDIT COMPANY

Smart Contract Security Audit

TechRate

November, 2021

Audit Details

Audited project

Invictus

Deployer address

0xD4710FFE134872183395Da547eE06c69E4acF033

Client contacts:

Invictus team

Blockchain

Ethereum

Project website:

www.invictustoken.com

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 Invictus to perform an audit of smart contracts:

https://etherscan.io/address/0xeb2ed9a5c7a8491b4faf987196baa50ee0855241#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

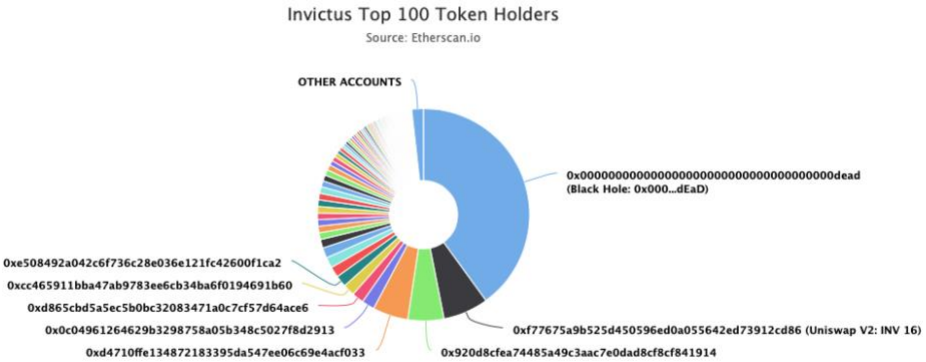
Token contract details for 28.11.2021

Contract name	Invictus
Contract address	0xeb2ed9a5c7A8491b4FaF987196BaA50EE0855241
Total supply	1,000,000,000,000,000
Token ticker	INV
Decimals	9
Token holders	131
Transactions count	383
Top 100 holders dominance	98.22%
Uniswap V2 pair	0xF77675a9B525d450596ed0a055642Ed73912CD86
Contract deployer address	0xD4710FFE134872183395Da547eE06c69E4acF033
Contract's current owner address	0x0000000000000000000000000000000000000000

Invictus Token Distribution

The top 100 holders collectively own 98.22% (982,228,916,357,274.00 Tokens) of Invictus

Token Total Supply: 1,000,000,000,000,000.00 Token | Total Token Holders: 131

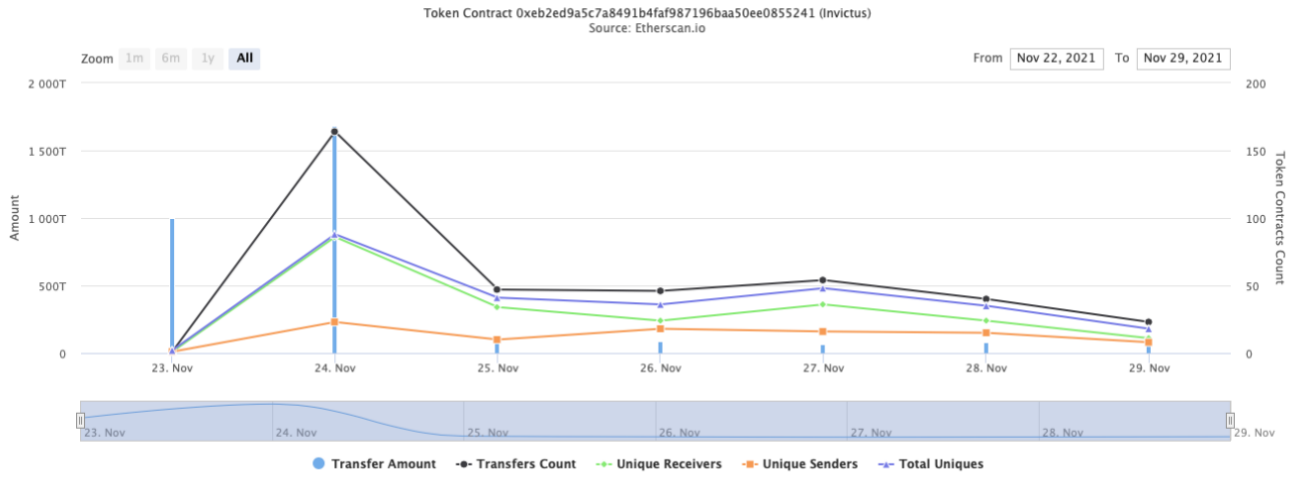


(A total of 982,228,916,357,274.00 tokens held by the top 100 accounts from the total supply of 1,000,000,000,000,000.00 token)

Invictus Contract Interaction Details

Time Series: Token Contract Overview

Tue 23, Nov 2021 - Mon 29, Nov 2021



Invictus Top 10 Token Holders

Rank	Address	Quantity (Token)	Percentage
1	Black Hole: 0x000...dEaD	400,000,000,000,000	40.0000%
2	Uniswap V2: INV 16	68,963,522,475,253.214564781	6.8964%
3	0x920d8cfea74485a49c3aac7e0dad8cf8cf841914	55,017,252,822,343.080078125	5.5017%
4	0xd4710ffe134872183395da547ee06c69e4acf033	54,000,000,000,000.000404738	5.4000%
5	0x0c04961264629b3298758a05b348c5027f8d2913	19,512,038,705,091.016583895	1.9512%
6	0xd865cbd5a5ec5b0bc32083471a0c7cf57d64ace6	18,018,981,728,945.874244066	1.8019%
7	0xcc465911bba47ab9783ee6cb34ba6f0194691b60	18,013,834,709,096.764704773	1.8014%
8	0xe508492a042c6f736c28e036e121fc42600f1ca2	17,575,198,923,230.170964941	1.7575%
9	0x9abc70916488f6bc02e9408a69e6d77691993659	16,284,078,672,232.145560577	1.6284%
10	0x2e58c60787d0178676a9a024e89dc305f47358da	16,158,666,179,487.93038476	1.6159%

Contract functions details

+ Context
  - [Int] _msgSender

+ [Int] IERC20
  - [Ext] totalSupply
  - [Ext] balanceOf
  - [Ext] transfer #
  - [Ext] allowance
  - [Ext] approve #
  - [Ext] transferFrom #

+ Ownable (Context)
  - [Pub] <Constructor> #
  - [Pub] owner
  - [Pub] renounceOwnership #
    - modifiers: onlyOwner
  - [Pub] transferOwnership #
    - modifiers: onlyOwner

+ [Lib] SafeMath
  - [Int] add
  - [Int] sub
  - [Int] sub
  - [Int] mul
  - [Int] div
  - [Int] div

+ [Int] IUniswapV2Factory
  - [Ext] createPair #

+ [Int] IUniswapV2Router02
  - [Ext] swapExactTokensForETHSupportingFeeOnTransferTokens #
  - [Ext] factory
  - [Ext] WETH
  - [Ext] addLiquidityETH ($)

+ Invictus (Context, IERC20, Ownable)
  - [Pub] <Constructor> #
  - [Pub] name
  - [Pub] symbol
  - [Pub] decimals
  - [Pub] totalSupply
  - [Pub] balanceOf
  - [Pub] transfer #
  - [Pub] allowance
  - [Pub] approve #
  - [Pub] transferFrom #
  - [Prv] tokenFromReflection
  - [Prv] removeAllFee #
  - [Prv] restoreAllFee #
  - [Prv] _approve #

- [Prv] _transfer #
- [Prv] swapTokensForEth #
  - modifiers: lockTheSwap
- [Prv] sendETHToFee #
- [Pub] setTrading #
  - modifiers: onlyOwner
- [Ext] manualswap #
- [Ext] manualsend #
- [Pub] blockBots #
  - modifiers: onlyOwner
- [Pub] unblockBot #
  - modifiers: onlyOwner
- [Prv] _tokenTransfer #
- [Prv] _transferStandard #
- [Prv] _takeTeam #
- [Prv] _reflectFee #
- [Ext] <Fallback> ($)
- [Prv] _getValues
- [Prv] _getTValues
- [Prv] _getRValues
- [Prv] _getRate
- [Prv] _getCurrentSupply
- [Pub] setFee #
  - modifiers: onlyOwner
- [Pub] setMinSwapTokensThreshold #
  - modifiers: onlyOwner
- [Pub] toggleSwap #
  - modifiers: onlyOwner
- [Pub] setMaxTxnAmount #
  - modifiers: onlyOwner
- [Pub] setMaxWalletSize #
  - modifiers: onlyOwner
- [Pub] excludeMultipleAccountsFromFees #
  - modifiers: onlyOwner
- [Pub] allowPreTrading #
  - modifiers: onlyOwner

($) = payable function
= non-constant function

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.	Passed
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

No medium severity issues found.

Low Severity Issues

No low severity issues found.

Owner privileges (In the period when the owner is not renounced)

- Owner can open / close trading.

```
function setTrading(bool _tradingOpen) public onlyOwner {
    tradingOpen = _tradingOpen;
}
```

- Owner can add and remove bots (no transferring between this addresses).

```
function blockBots(address[] memory bots_) public onlyOwner {
    for (uint256 i = 0; i < bots_.length; i++) {
        bots[bots_[i]] = true;
    }
}

function unblockBot(address notbot) public onlyOwner {
    bots[notbot] = false;
}
```

- Owner can change redis and tax fees.

```
function setFee(
    uint256 redisFeeOnBuy,
    uint256 redisFeeOnSell,
    uint256 taxFeeOnBuy,
    uint256 taxFeeOnSell
) public onlyOwner {
    _redisFeeOnBuy = redisFeeOnBuy;
    _redisFeeOnSell = redisFeeOnSell;

    _taxFeeOnBuy = taxFeeOnBuy;
    _taxFeeOnSell = taxFeeOnSell;
}
```

- Owner can change minimum swap tokens at amount value.

```
function setMinSwapTokensThreshold(uint256 swapTokensAtAmount)
    public
    onlyOwner
{
    _swapTokensAtAmount = swapTokensAtAmount;
}
```

- Owner can enable / disable swap.

```
function toggleSwap(bool _swapEnabled) public onlyOwner {
    swapEnabled = _swapEnabled;
}
```

- Owner can change maximum transaction limit.

```
function setMaxTxnAmount(uint256 maxTxAmount) public onlyOwner {
    _maxTxAmount = maxTxAmount;
}
```

- Owner can change maximum token per wallet.

```
function setMaxWalletSize(uint256 maxWalletSize) public onlyOwner {
    _maxWalletSize = maxWalletSize;
}
```

- Owner can include in and exclude from fees.

```
function excludeMultipleAccountsFromFees(
    address[] calldata accounts,
    bool excluded
) public onlyOwner {
    for (uint256 i = 0; i < accounts.length; i++) {
        _isExcludedFromFee[accounts[i]] = excluded;
    }
}
```

- Owner can enable / disable pre trading for addresses.

```
function allowPreTrading(address account, bool allowed) public onlyOwner {
    require(preTrader[account] != allowed, "TOKEN: Already enabled.");
    preTrader[account] = allowed;
}
```

- Development address can manually swap.

```
function manualswap() external {
    require(_msgSender() == _developmentAddress);
    uint256 contractBalance = balanceOf(address(this));
    swapTokensForEth(contractBalance);
}
```

- Development address can manually withdraw contract ETHs.

```
function manualsend() external {
    require(_msgSender() == _developmentAddress);
    uint256 contractETHBalance = address(this).balance;
    sendETHToFee(contractETHBalance);
}
```

Conclusion

Smart contracts do not contain high severity issues! Liquidity pair contract's security is not checked due to out of scope.

Liquidity locking details provided by the team:
https://www.team.finance/view-coin/0xeb2ed9a5c7A8491b4FaF987196BaA50EE0855241?name=Invictus&symbol=INV

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.