



Coinsult

Advanced Manual Smart Contract Audit

CRONEX

Project: Cronex

Website: <https://cronex.finance/>

Low-Risk

2 low-risk code
issues found

Medium-Risk

0 medium-risk code
issues found

High-Risk

0 high-risk code
issues found

Contract Address

Contract not deployed yet

Disclaimer: Coinsult is not responsible for any financial losses. Nothing in this contract audit is financial advice, please do your own research.

Disclaimer

Coinsult is not responsible if a project turns out to be a scam, rug-pull or honeypot. We only provide a detailed analysis for your own research.

Coinsult is not responsible for any financial losses. Nothing in this contract audit is financial advice, please do your own research.

The information provided in this audit is for informational purposes only and should not be considered investment advice. Coinsult does not endorse, recommend, support or suggest to invest in any project.

Coinsult can not be held responsible for when a project turns out to be a rug-pull, honeypot or scam.

Tokenomics

Contract not deployed yet

Source Code

Coinsult was commissioned by Cronex to perform an audit based on the following smart contract:

<https://github.com/cronexfinance/cronex-token/blob/main/CRONEX.sol>

Manual Code Review

In this audit report we will highlight all these issues:

Low-Risk

2 low-risk code
issues found

Medium-Risk

0 medium-risk code
issues found

High-Risk

0 high-risk code
issues found

The detailed report continues on the next page...

● **Low-Risk:** Could be fixed, will not bring problems.

Avoid relying on `block.timestamp`

`block.timestamp` can be manipulated by miners.

```
function swapAndSendToFee(uint256 amount) private {
    cronexRouter.swapExactTokensForETHSupportingFeeOnTransferTokens(
        amount,
        0,
        path,
        marketingWalletAddress,
        block.timestamp
    );
}
```

Recommendation

Do not use `block.timestamp`, `now` or `blockhash` as a source of randomness

Exploit scenario

```
contract Game {

    uint reward_determining_number;

    function guessing() external{
        reward_determining_number = uint256(block.blockhash(10000)) % 10;
    }
}
```

Eve is a miner. Eve calls `guessing` and re-orders the block containing the transaction. As a result, Eve wins the game.

● **Low-Risk:** Could be fixed, will not bring problems.

Missing events arithmetic

Detect missing events for critical arithmetic parameters.

```
function setMaxBalanceAmount(uint256 maxBalanceAmount) public onlyOwner {
    require(maxBalanceAmount >= totalSupply / 100, "CRONEX: Max Wallet cannot be below 1%"); // 1
    maxBalance = maxBalanceAmount;
}
```

Recommendation

Emit an event for critical parameter changes.

Exploit scenario

```
contract C {

    modifier onlyAdmin {
        if (msg.sender != owner) throw;
        _;
    }

    function updateOwner(address newOwner) onlyAdmin external {
        owner = newOwner;
    }
}
```

updateOwner() has no event, so it is difficult to track off-chain changes in the buy price.

Owner privileges

- Owner cannot set buy and sell fee higher than 6%
- Owner cannot set max balance amount lower than 1%
- Owner cannot pause trading
- Owner can exclude from fees
- Owner can change max transaction amount (Max Transaction cannot be below 0.25%)

Extra notes by the team

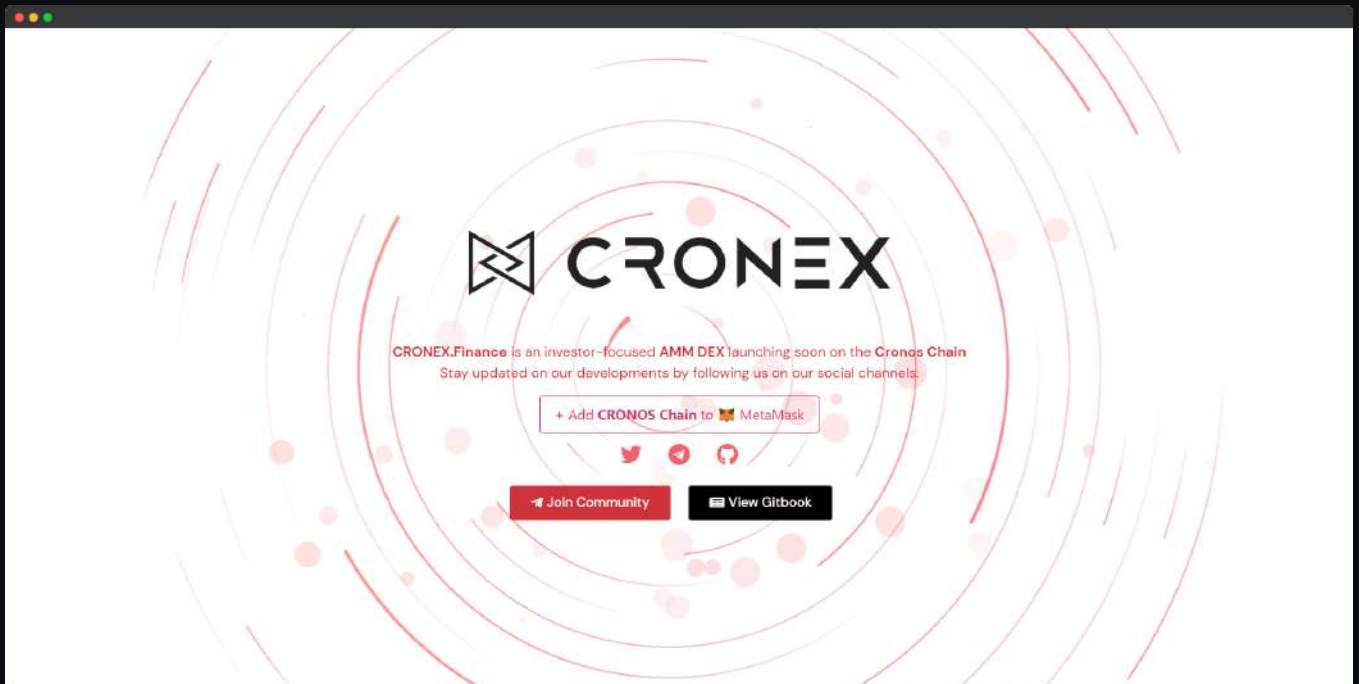
The usage of `block.timestamp` is only needed as a deadline to complete the transaction and not as a source of randomness.

Contract Snapshot

```
contract CRONEXToken is ERC20, Ownable {  
    string public constant override name = "CRONEX";  
    string public constant override symbol = "CRONEX";  
    uint8 public constant override decimals = 18;  
    uint256 public override totalSupply = 100_000 * 1e18;  
    mapping(address => mapping(address => uint256)) public override allowance;
```

Website Review

Coinsult checks the website completely manually and looks for visual, technical and textual errors. We also look at the security, speed and accessibility of the website. In short, a complete check to see if the website meets the current standard of the web development industry.



- Mobile Friendly
- Does not contain jQuery errors
- SSL Secured
- No major spelling errors

Project Overview

 Not KYC verified by Coinsult

AUDITED
BY COINSULT.NET

CRONEX

