July 12, 2023

Express Audit Report for

# FunValueCoin [FVC]



De.Fi

De.Fi

# Project Summary

| | |
|---|---|
| Project Name | FunValueCoin |
| Address | 0x576E4FdB544bB86b89988cAEDD4D0aE3582A9943 |
| Network | 1 |

De.Fi

| | |
|---|---|
| Issue ID | 183 |
| Severity | High |
| Status | Optimization |
| Description Code | **uint256 private** _totalSupply = 1000000000 * 10**18; |
| Location | FunValueCoin._totalSupply (erctoken.sol#43) should be constant |

De.Fi

| | |
|---|---|
| Issue ID | 183 |
| Severity | High |
| Status | Optimization |
| Description Code | **uint256 public** burnTax = 1; |
| Location | FunValueCoin.burnTax (erctoken.sol#33) should be constant |

De.Fi

| | |
|---|---|
| Issue ID | 183 |
| Severity | High |
| Status | Optimization |
| Description Code | **uint256 public** marketingTax = 1; |
| Location | FunValueCoin.marketingTax (erctoken.sol#32) should be constant |

De.Fi

| | |
|---|---|
| Issue ID | 183 |
| Severity | High |
| Status | Optimization |
| Description Code | `uint256 public rewardTax = 1;` |
| Location | FunValueCoin.rewardTax (erctoken.sol#31) should be constant |

| | |
|---|---|
| Issue ID | 103 |
| Severity | High |
| Status | Informational |
| Description Code | **pragma solidity** ^0.8.0; |
| Location | Different versions of Solidity is used:<br>- Version used: ['^0.8.0', '^0.8.1']<br>- ^0.8.0 (AccessControl.sol#4)<br>- ^0.8.0 (IAccessControl.sol#4)<br>- ^0.8.0 (Ownable.sol#4)<br>- ^0.8.0 (Pausable.sol#4)<br>- ^0.8.0 (ReentrancyGuard.sol#4)<br>- ^0.8.0 (ERC20.sol#4)<br>- ^0.8.0 (IERC20.sol#4)<br>- ^0.8.0 (IERC20Metadata.sol#4)<br>- ^0.8.0 (IERC20Permit.sol#4)<br>- ^0.8.0 (SafeERC20.sol#4)<br>- ^0.8.1 (Address.sol#4)<br>- ^0.8.0 (Context.sol#4)<br>- ^0.8.0 (Strings.sol#4)<br>- ^0.8.0 (ERC165.sol#4)<br>- ^0.8.0 (IERC165.sol#4)<br>- ^0.8.0 (Math.sol#4)<br>- ^0.8.0 (SafeMath.sol#4)<br>- ^0.8.0 (SignedMath.sol#4)<br>- ^0.8.0 (AccessControl.sol#4)<br>- ^0.8.1 (Address.sol#4)<br>- ^0.8.0 (Context.sol#4)<br>- ^0.8.0 (ERC165.sol#4)<br>- ^0.8.0 (ERC20.sol#4)<br>- ^0.8.0 (IAccessControl.sol#4)<br>- ^0.8.0 (IERC165.sol#4)<br>- ^0.8.0 (IERC20.sol#4)<br>- ^0.8.0 (IERC20Metadata.sol#4)<br>- ^0.8.0 (IERC20Permit.sol#4)<br>- ^0.8.0 (Math.sol#4)<br>- ^0.8.0 (Ownable.sol#4)<br>- ^0.8.0 (Pausable.sol#4)<br>- ^0.8.0 (ReentrancyGuard.sol#4)<br>- ^0.8.0 (SafeERC20.sol#4) |

De.Fi

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |
| Location | Math.mulDiv(uint256,uint256,uint256) (Math.sol#55-134) performs a multiplication on the result of a division:<br>-denominator = denominator / twos (Math.sol#101)<br>-inverse = (3 * denominator) ^ 2 (Math.sol#116) |

De.Fi

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |
| Location | Math.mulDiv(uint256,uint256,uint256) (Math.sol#55-134) performs a multiplication on the result of a division:<br>-denominator = denominator / twos (Math.sol#101)<br>-inverse *= 2 - denominator * inverse (Math.sol#120) |

De.Fi

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |
| Location | Math.mulDiv(uint256,uint256,uint256) (Math.sol#55-134) performs a multiplication on the result of a division:<br>-denominator = denominator / twos (Math.sol#101)<br>-inverse *= 2 - denominator * inverse (Math.sol#121) |

De.Fi

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |
| Location | Math.mulDiv(uint256,uint256,uint256) (Math.sol#55-134) performs a multiplication on the result of a division:<br>-denominator = denominator / twos (Math.sol#101)<br>-inverse *= 2 - denominator * inverse (Math.sol#122) |

De.Fi

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |
| Location | Math.mulDiv(uint256,uint256,uint256) (Math.sol#55-134) performs a multiplication on the result of a division:<br>-denominator = denominator / twos (Math.sol#101)<br>-inverse *= 2 - denominator * inverse (Math.sol#123) |

De.Fi

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |
| Location | Math.mulDiv(uint256,uint256,uint256) (Math.sol#55-134) performs a multiplication on the result of a division:<br>-denominator = denominator / twos (Math.sol#101)<br>-inverse *= 2 - denominator * inverse (Math.sol#124) |

De.Fi

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |
| Location | Math.mulDiv(uint256,uint256,uint256) (Math.sol#55-134) performs a multiplication on the result of a division:<br>-denominator = denominator / twos (Math.sol#101)<br>-inverse *= 2 - denominator * inverse (Math.sol#125) |

De.Fi

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |
| Location | Math.mulDiv(uint256,uint256,uint256) (Math.sol#55-134) performs a multiplication on the result of a division:<br>-prod0 = prod0 / twos (Math.sol#104)<br>-result = prod0 * inverse (Math.sol#131) |

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |

```
function _transfer(address sender, address recipient, uint256 amount) internal override nonReentrant whenNotPaused {
require(sender != address(0) && recipient != address(0), "Cannot transfer from/to zero address");
require(rewardTax.add(marketingTax).add(burnTax) <= MAX_TAX, "Sum of taxes must not exceed 5");
uint256 sendAmount = amount;
if (isTaxEnabled && !isTaxExempt[sender]) {
uint256 taxSum = rewardTax.add(marketingTax).add(burnTax);
uint256 taxAmount = amount.mul(taxSum).div(PERCENT_DENOMINATOR);
sendAmount = amount.sub(taxAmount);
uint256 rewardTaxAmount = taxAmount.mul(rewardTax).div(taxSum);
uint256 marketingTaxAmount = taxAmount.mul(marketingTax).div(taxSum);
uint256 burnTaxAmount = taxAmount.mul(burnTax).div(taxSum);
// Reward Tax
super._transfer(sender, rewardAddress, rewardTaxAmount);
emit TaxTransferred(rewardAddress, rewardTaxAmount);
// Marketing Tax
super._transfer(sender, marketingAddress, marketingTaxAmount);
emit TaxTransferred(marketingAddress, marketingTaxAmount);
// Burn Tax
_burn(sender, burnTaxAmount);
}
super._transfer(sender, recipient, sendAmount);
}
```

| | |
|---|---|
| Location | FunValueCoin._transfer(address,address,uint256)<br><br>(arctoken.sol#94-122) performs a multiplication on |

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |

```
function _transfer(address sender, address recipient, uint256 amount) internal override nonReentrant whenNotPaused {
require(sender != address(0) && recipient != address(0), "Cannot transfer from/to zero address");
require(rewardTax.add(marketingTax).add(burnTax) <= MAX_TAX, "Sum of taxes must not exceed 5");
uint256 sendAmount = amount;
if (isTaxEnabled && !isTaxExempt[sender]) {
uint256 taxSum = rewardTax.add(marketingTax).add(burnTax);
uint256 taxAmount = amount.mul(taxSum).div(PERCENT_DENOMINATOR);
sendAmount = amount.sub(taxAmount);
uint256 rewardTaxAmount = taxAmount.mul(rewardTax).div(taxSum);
uint256 marketingTaxAmount = taxAmount.mul(marketingTax).div(taxSum);
uint256 burnTaxAmount = taxAmount.mul(burnTax).div(taxSum);
// Reward Tax
super._transfer(sender, rewardAddress, rewardTaxAmount);
emit TaxTransferred(rewardAddress, rewardTaxAmount);
// Marketing Tax
super._transfer(sender, marketingAddress, marketingTaxAmount);
emit TaxTransferred(marketingAddress, marketingTaxAmount);
// Burn Tax
_burn(sender, burnTaxAmount);
}
super._transfer(sender, recipient, sendAmount);
}
```

| | |
|---|---|
| Location | FunValueCoin._transfer(address,address,uint256) |

| | |
|---|---|
| Issue ID | 156 |
| Severity | Medium |
| Status | Low |
| Description Code | |

```
function _transfer(address sender, address recipient, uint256 amount) internal override nonReentrant whenNotPaused {
require(sender != address(0) && recipient != address(0), "Cannot transfer from/to zero address");
require(rewardTax.add(marketingTax).add(burnTax) <= MAX_TAX, "Sum of taxes must not exceed 5");
uint256 sendAmount = amount;
if (isTaxEnabled && !isTaxExempt[sender]) {
uint256 taxSum = rewardTax.add(marketingTax).add(burnTax);
uint256 taxAmount = amount.mul(taxSum).div(PERCENT_DENOMINATOR);
sendAmount = amount.sub(taxAmount);
uint256 rewardTaxAmount = taxAmount.mul(rewardTax).div(taxSum);
uint256 marketingTaxAmount = taxAmount.mul(marketingTax).div(taxSum);
uint256 burnTaxAmount = taxAmount.mul(burnTax).div(taxSum);
// Reward Tax
super._transfer(sender, rewardAddress, rewardTaxAmount);
emit TaxTransferred(rewardAddress, rewardTaxAmount);
// Marketing Tax
super._transfer(sender, marketingAddress, marketingTaxAmount);
emit TaxTransferred(marketingAddress, marketingTaxAmount);
// Burn Tax
_burn(sender, burnTaxAmount);
}
super._transfer(sender, recipient, sendAmount);
}
```

| | |
|---|---|
| Location | FunValueCoin._transfer(address,address,uint256) |

(erc20.sol#94-132) performs a multiplication on

De.Fi

| | |
|---|---|
| Issue ID | 177 |
| Severity | High |
| Status | Informational |
| Description Code | **pragma solidity** ^0.8.0; |
| Location | Pragma version^0.8.0 (AccessControl.sol#4) allows old versions |

De.Fi

| | |
|---|---|
| Issue ID | 177 |
| Severity | High |
| Status | Informational |
| Description Code | **pragma solidity** ^0.8.1; |
| Location | Pragma version^0.8.1 (Address.sol#4) allows old versions |

| | |
|---|---|
| Issue ID | 177 |
| Severity | High |
| Status | Informational |
| Description Code | |
| Location | solc-0.8.1 is not recommended for deployment |

De.Fi

| | |
|---|---|
| Issue ID | 173 |
| Severity | High |
| Status | Informational |
| Description Code | **function _callOptionalReturnBool**(IERC20 token, **bytes memory** data) **private returns** (**bool**) {<br>// We need to perform a low level call here, to bypass Solidity's return data size checking mechanism, since<br>// we're implementing it ourselves. We cannot use {Address-functionCall} here since this should return false<br>// and not revert is the subcall reverts.<br>(**bool** success, **bytes memory** returndata) = **address**(token).call(data);<br>**return**<br>success && (returndata.length == 0 \|\| abi.decode(returndata, (**bool**))) && Address.isContract(**address**(token));<br>} |
| Location | Low level call in SafeERC20._callOptionalReturnBool(IERC20,bytes) (SafeERC20.sol#134-142):<br>- (success,returndata) = address(token).call(data) (SafeERC20.sol#139) |

De.Fi

| | |
|---|---|
| Issue ID | 173 |
| Severity | High |
| Status | Informational |
| Description Code | **function sendValue**(**address payable** recipient, **uint256** amount) **internal** {<br>require(**address**(this).balance >= amount, "Address: insufficient balance");<br>(**bool** success, ) = recipient.call{value: amount}("");<br>require(success, "Address: unable to send value, recipient may have reverted");<br>} |
| Location | Low level call in Address.sendValue(address,uint256) (Address.sol#64-69):<br>- (success) = recipient.call{value: amount}() (Address.sol#67) |

De.Fi

| | |
|---|---|
| Issue ID | 173 |
| Severity | High |
| Status | Informational |
| Description Code | **function functionCallWithValue**(<br>**address** target,<br>**bytes memory** data,<br>**uint256** value,<br>**string memory** errorMessage<br>) **internal returns** (**bytes memory**) {<br>require(**address**(this).balance >= value, "Address: insufficient balance for call");<br>(**bool** success, **bytes memory** returndata) = target.call{value: value}(data);<br>**return** verifyCallResultFromTarget(target, success, returndata, errorMessage);<br>} |
| Location | Low level call in Address.functionCallWithValue(address,bytes,uint256,string) (Address.sol#128-137):<br>- (success,returndata) = target.call{value: value}(data) (Address.sol#135) |

De.Fi

| | |
|---|---|
| Issue ID | 173 |
| Severity | High |
| Status | Informational |
| Description Code | **function functionStaticCall**(<br>**address** target,<br>**bytes memory** data,<br>**string memory** errorMessage<br>) **internal view returns** (**bytes memory**) {<br>(**bool** success, **bytes memory** returndata) =<br>target.staticcall(data);<br>**return** verifyCallResultFromTarget(target, success,<br>returndata, errorMessage);<br>} |
| Location | Low level call in<br>Address.functionStaticCall(address,bytes,string)<br>(Address.sol#155-162):<br>- (success,returndata) = target.staticcall(data)<br>(Address.sol#160) |

De.Fi

| | |
|---|---|
| Issue ID | 173 |
| Severity | High |
| Status | Informational |
| Description Code | **function functionDelegateCall(**<br>**address** target,<br>**bytes memory** data,<br>**string memory** errorMessage<br>) **internal returns** (**bytes memory**) {<br>(**bool** success, **bytes memory** returndata) =<br>target.delegatecall(data);<br>**return** verifyCallResultFromTarget(target, success,<br>returndata, errorMessage);<br>} |
| Location | Low level call in<br>Address.functionDelegateCall(address,bytes,string)<br>(Address.sol#180-187):<br>- (success,returndata) = target.delegatecall(data)<br>(Address.sol#185) |

De.Fi

| | |
|---|---|
| Issue ID | 189 |
| Severity | High |
| Status | Critical |
| Description Code | |
| Location | Pausable function: ERC20.transfer(address,uint256) (ERC20.sol#113-117)<br>- in internal call:_transfer<br>In modifier:whenNotPaused<br>- In expression: require(bool,string)(!paused(),Pausable: paused) |

De.Fi

| | |
|---|---|
| Issue ID | 189 |
| Severity | High |
| Status | Critical |
| Description Code | |
| Location | Pausable function:<br>ERC20.transferFrom(address,address,uint256)<br>(ERC20.sol#158-163)<br>- in internal call:_transfer<br>In modifier:whenNotPaused<br>- In expression: require(bool,string)(!<br>paused(),Pausable: paused) |

De.Fi

| | |
|---|---|
| Issue ID | 182 |
| Severity | Medium |
| Status | Informational |
| Description Code | **uint256 private** _totalSupply = 1000000000 * 10**18; |
| Location | Contract FunValueCoin uses literals with too many digits:<br>- _totalSupply = 1000000000 * 10 ** 18 (erctoken.sol#43) |

De.Fi

| | |
|---|---|
| Issue ID | 209 |
| Severity | High |
| Status | Critical |
| Description Code | |
| Location | Transfer Fee: ERC20.transfer(address,uint256) (ERC20.sol#113-117)<br>- in nested function: _transfer<br>- in expression: amount.mul(taxSum).div(PERCENT_DENOMINATOR)<br>- in expression: rewardTax.add(marketingTax).add(burnTax)<br>- in expression: taxAmount.mul(rewardTax).div(taxSum)<br>- in expression: taxAmount.mul(marketingTax).div(taxSum) |

| | |
|---|---|
| Issue ID | 7 |
| Severity | High |
| Status | Data |
| Description Code | |
| Location | Transfer fee variables |