

ERC20 Verifier

Enter the address of an ERC20 contract to check
if it conforms to the standard

0xA42bE035918A91406Ad0d756e96af0dD442E12Df

VERIFY

Contract MiniMeToken

== ERC20 functions definition ==

[✓] transfer (address, uint256) -> (bool)
[✓] approve (address, uint256) -> (bool)
[✓] transferFrom (address, address, uint256) -> (bool)
[✓] allowance (address, address) -> (uint256)
[✓] balanceOf (address) -> (uint256)

== Custom modifiers ==

[✓] No custom modifiers in ERC20 functions

== ERC20 events ==

[✓] Transfer (address, address, uint256)
[✓] Approval (address, address, uint256)
[✓] transfer must emit Transfer (address, address, uint256)
[✓] approve must emit Approval (address, address, uint256)
[✓] transferFrom must emit Transfer (address, address, uint256)

== ERC20 getters ==

[✓] totalSupply () -> (uint256)
[✓] decimals () -> (uint8)
[✓] symbol () -> (string)
[✓] name () -> (string)

== Allowance frontrunning mitigation ==

[x] increaseAllowance (address, uint256) -> (bool)
[x] decreaseAllowance (address, uint256) -> (bool)

== Balance check in approve function ==

[✓] approve function should not check for sender's balance

The contract source code must be verified on etherscan, and be compiled with Solidity v0.4 or v0.5. This page uses [tinchoabbate's slither-scripts](#) to check whether a contract is a valid ERC20 or not using Slither. The script does not verify that the functions found behave as expected. It just checks for matching signatures, return types, existence of custom modifiers, event emissions, among others. These scripts may have bugs, use them at your own risk.

Have any feedback? Share any thoughts in the [OpenZeppelin forum](#).