|  |  |  |  |
| --- | --- | --- | --- |
| 1 | ERC20\_event1.sol | 检测ERC20\_event1.sol智能合约，验证应当通过  （该智能合约中插入了多条emits约束均被满足，验证应当通过） | 1. 打开网页  https://github.com/my-code-cloud/EvaluationCases/tree/main/solidity%E5%90%88%E7%BA%A6%E5%BD%A2%E5%BC%8F%E5%8C%96%E9%AA%8C%E8%AF%81  2. 选择ERC20\_event1.sol将其代码复制到<http://39.103.152.161/>的合约代码中  3. 选择Solidity，形式化验证  4. 点击Analyze Now  5. 验证通过 |
| 2 | ERC20\_event2.sol | 检测ERC20\_event2.sol智能合约，验证应当不通过  （该智能合约中插入了多条emits约束，其中一条加在transfer函数上的emits Transfer约束指transfer函数必须产生Transfer日志，但是该函数并没有调用emit功能产生日志，验证应当不通过） | 1. 打开网页  https://github.com/my-code-cloud/EvaluationCases/tree/main/solidity%E5%90%88%E7%BA%A6%E5%BD%A2%E5%BC%8F%E5%8C%96%E9%AA%8C%E8%AF%81  2. 选择ERC20\_event2.sol将其代码复制到<http://39.103.152.161/>的合约代码中  3. 选择Solidity，形式化验证  4. 点击Analyze Now  5. 验证不通过 |
| 3 | ERC20\_totalsuppply1.sol | 检测ERC20\_totalsuppply1.sol智能合约，验证应当通过  （\_\_verifier\_sum\_uint(balances) == totalSupply指balances数据结构的int类型的值的总和应该和totalSupply保持相等，该约束被放在了合约边外，对合约内所有函数有效，任意一个transfer函数均是对balances组内的转账，验证应当通过） | 1. 打开网页  https://github.com/my-code-cloud/EvaluationCases/tree/main/solidity%E5%90%88%E7%BA%A6%E5%BD%A2%E5%BC%8F%E5%8C%96%E9%AA%8C%E8%AF%81  2. 选择ERC20\_totalsuppply1.sol将其代码复制到<http://39.103.152.161/>的合约代码中  3. 选择Solidity，形式化验证  4. 点击Analyze Now  5. 验证通过 |
| 4 | ERC20\_totalsuppply2.sol | 检测ERC20\_totalsuppply2.sol智能合约，验证应当不通过  （\_\_verifier\_sum\_uint(balances) == totalSupply指balances数据结构的int类型的值的总和应该和totalSupply保持相等，该约束被放在了合约边外，对合约内所有函数有效，其中一个transfer函数是对balances组内的转账，但是仅有付款的语句没有收款的语句，balances总数减少，验证应当不通过） | 1. 打开网页  https://github.com/my-code-cloud/EvaluationCases/tree/main/solidity%E5%90%88%E7%BA%A6%E5%BD%A2%E5%BC%8F%E5%8C%96%E9%AA%8C%E8%AF%81  2. 选择ERC20\_totalsuppply2.sol将其代码复制到<http://39.103.152.161/>的合约代码中  3. 选择Solidity，形式化验证  4. 点击Analyze Now  5. 验证不通过 |
| 5 | ERC20\_modifies1.sol | 检测ERC20\_modifies1.sol智能合约，验证应当通过  （该智能合约中插入了多条modifies约束均被满足，验证应当通过） | 1. 打开网页  https://github.com/my-code-cloud/EvaluationCases/tree/main/solidity%E5%90%88%E7%BA%A6%E5%BD%A2%E5%BC%8F%E5%8C%96%E9%AA%8C%E8%AF%81  2. 选择ERC20\_modifies1.sol将其代码复制到<http://39.103.152.161/>的合约代码中  3. 选择Solidity，形式化验证  4. 点击Analyze Now  5. 验证通过 |
| 6 | ERC20\_modifies2.sol | 检测ERC20\_modifies2.sol智能合约，验证应当不通过  （该智能合约中插入了多条modifie条件，其中一条modifies balanceOf if balanceOf[msg.sender] >= amount指只有当balanceOf[msg.sender] >= amount时才能修改balanceOf，但是transfer函数中没有对balanceOf[msg.sender] >= amount的require要求，导致可以对balanceOf无条件修改，该约束条件失效，验证应当不通过） | 1. 打开网页  https://github.com/my-code-cloud/EvaluationCases/tree/main/solidity%E5%90%88%E7%BA%A6%E5%BD%A2%E5%BC%8F%E5%8C%96%E9%AA%8C%E8%AF%81  2. 选择ERC20\_modifies2.sol将其代码复制到<http://39.103.152.161/>的合约代码中  3. 选择Solidity，形式化验证  4. 点击Analyze Now  5. 验证不通过 |