Token Contrac Review

이진호

컨트랙트 상태변수

- 토큰 기본정보를 저장하기위한 상태변수

Token MyToken888

Sponsored: MythX Smart contract security analysis, Pro options for mission-critical dapps. Start analyzing now

Overview [ERC-20]	
Total Supply:	100,000 MT8
Holders:	1 addresses
Transfers:	1

Profile Summary [Edit]	
Contract:	0x7dba38ded725f4d98b89ccf8b171dd39515ab4fc
Decimals:	18

```
string public name;
string public symbol;
uint8 public decimals;

mapping (address => uint256) private _balances;
mapping (address => mapping (address => uint256)) private _allowed;

uint256 private _totalSupply;
```

컨트랙트 상태변수

- **_balance**
  - 각 주소별로 토큰 잔액정보를 저장하기 위한 테이블
  - Private 옵션을 통해 외부에서 상태변수를 직접 볼 수 없도록 함

```
string public name;
string public symbol;
uint8 public decimals;

mapping (address => uint256) private _balances;
mapping (address => mapping (address => uint256)) private _allowed;

uint256 private _totalSupply;
```

컨트랙트 상태변수

Token MyToken888

Sponsored: MythX Smart contract security analysis, Pro options for mission-critical dapps. Start analyzing now

Overview [ERC-20]

Total Supply:	100,000 MT8
Holders:	1 addresses
Transfers:	1

Profile Summary [Edit]

Contract:	0x7dba38ded725f4d98b89ccf8b171dd39515ab4fc
Decimals:	18

Transfers | Holders | Read Contract | Write Contract

Token Holders Chart

A total of 1 token holder

First | Page 1 of 1 | Last

Rank	Address	Quantity	Percentage
1	0x8861862af614bf0ff1a8dde3200a5a370e88be38	100,000	100.0000%

```
mapping (address => uint256) private _balances;
```

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	100,000

컨트랙트 상태변수

- **_allowed**
  - 각 주소별로 제 3자에게 토큰사용을 허락한 정보
  - mapping 안의 Value가 또다른 mapping 테이블
  - Private 옵션을 통해 외부에서 상태변수를 직접 볼 수 없도록 함

```
string public name;
string public symbol;
uint8 public decimals;

mapping (address => uint256) private _balances;
mapping (address => mapping (address => uint256)) private _allowed;

uint256 private _totalSupply;
```

컨트랙트 상태변수

```
mapping (address => mapping (address => uint256)) private _allowed;
```

_balances	
<KEY> address	<Value> mapping (address => uint256)
0x8861……8be38	mapping(address, uint256)

<KEY> address	<Value> uint256
0x42df……a5c6	100

Event

- **Event**
  - 트랜잭션 수행결과에 따라, 남기기 위한 로그 데이터
  - emit 이라는 명령어를 통해 트랜잭션 이력에 이벤트를 발생
  - 노드에서는 트랜잭션에서 이벤트 발생시, 이를 감지하여, 다른 프로세스 진행이 가능

    ex) 입,출금 이벤트에 따른 거래소 지갑 DB 업데이트

```
event Transfer(address indexed _from, address indexed _to, uint256 _value);
event Approval(address indexed _owner, address indexed _spender, uint256 _value);
```

Event

- **생성자**
  - 컨트랙트 배포에 따른 기본 값 설정
  - _mint 함수 호출을 통한 토큰 수량 생성

```
constructor() public{
     name = "MyToken888";
     symbol="MT8";
     decimals = 18;
     _mint(msg.sender,100000 * (10**18));
 }
```

Constant Function

- **Constant Function**
  - 토큰의 상태값 조회를 위한 Read-Only Function
  - Private 상태변수의 값 조회를 위한 getter

```
function totalSupply() public view returns (uint256) {
  return _totalSupply;
}

function balanceOf(address owner) public view returns (uint256) {
  return _balances[owner];
}

function allowance( address owner, address spender ) public view returns (uint256)
{
  return _allowed[owner][spender];
}
```

Constant Function

- **Constant Function**
  - 토큰의 상태값 조회를 위한 Read-Only Function
  - Private 상태변수의 값 조회를 위한 getter

```
function totalSupply() public view returns (uint256) {
  return _totalSupply;
}

function balanceOf(address owner) public view returns (uint256) {
  return _balances[owner];
}

function allowance( address owner, address spender ) public view returns (uint256)
{
  return _allowed[owner][spender];
}
```

Transfer

- **Transfer**
  - 토큰전송 기능
  - _transfer에서 실제 작업을 처리

```
function transfer(address to, uint256 value) public returns (bool) {
   _transfer(msg.sender, to, value);
   return true;
 }
```

Transfer

- **_transfer**
  - internal Function으로 외부에서 직접호출 불가
  - transfer / transferFrom을 통해서만 전송 가능
  - 토큰 전송 작업 전에 유효성 검사 실시
  - 작업 종료 후에 이벤트 발생

```
function _transfer(address from, address to, uint256 value) internal {
  require(value <= _balances[from]);
  require(to != address(0));

  _balances[from] = _balances[from].sub(value);
  _balances[to] = _balances[to].add(value);
  emit Transfer(from, to, value);
}
```

Transfer

- **_transfer**
  - internal Function으로 외부에서 직접호출 불가
  - transfer / transferFrom을 통해서만 전송 가능
  - 토큰 전송 작업 전에 유효성 검사 실시
  - 작업 종료 후에 이벤트 발생

```
function _transfer(address from, address to, uint256 value) internal {
  require(value <= _balances[from]);
  require(to != address(0));

  _balances[from] = _balances[from].sub(value);
  _balances[to] = _balances[to].add(value);
  emit Transfer(from, to, value);
}
```

Transfer

```
from : 0x8861…....8be38
to   : 0x42df…....a5c6
value: 100

function _transfer(address from, address to, uint256 value) internal {
   require(value <= _balances[from]);
   require(to != address(0));

   _balances[from] = _balances[from].sub(value);
   _balances[to] = _balances[to].add(value);
   emit Transfer(from, to, value);
 }
```

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	100,000

Transfer

```
from : 0x8861…...8be38
to   : 0x42df…...a5c6
value: 100

function _transfer(address from, address to, uint256 value) internal {
   require(value <= _balances[from]);
   require(to != address(0));

   _balances[from] = _balances[from].sub(value);
   _balances[to] = _balances[to].add(value);
   emit Transfer(from, to, value);
 }
```

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	100,000

Transfer

```
from : 0x8861…....8be38
to   : 0x42df…....a5c6
value: 100

function _transfer(address from, address to, uint256 value) internal {
   require(value <= _balances[from]);
   require(to != address(0));

   _balances[from] = _balances[from].sub(value);
   _balances[to] = _balances[to].add(value);
   emit Transfer(from, to, value);
 }
```

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	100,000

Transfer

```
from : 0x8861…....8be38
to   : 0x42df…....a5c6
value: 100

function _transfer(address from, address to, uint256 value) internal {
   require(value <= _balances[from]);
   require(to != address(0));

   _balances[from] = _balances[from].sub(value);
   _balances[to] = _balances[to].add(value);
   emit Transfer(from, to, value);
 }
```

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	99,900

Transfer

```
from : 0x8861…....8be38
to   : 0x42df…....a5c6
value: 100

function _transfer(address from, address to, uint256 value) internal {
   require(value <= _balances[from]);
   require(to != address(0));

   _balances[from] = _balances[from].sub(value);
   _balances[to] = _balances[to].add(value);
   emit Transfer(from, to, value);
 }
```

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	99,900
0	0

Transfer

```
from : 0x8861…...8be38
to   : 0x42df…...a5c6
value: 100

function _transfer(address from, address to, uint256 value) internal {
   require(value <= _balances[from]);
   require(to != address(0));

   _balances[from] = _balances[from].sub(value);
   _balances[to] = _balances[to].add(value);
   emit Transfer(from, to, value);
 }
```

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	99,900
0x42df……a5c6	0

Transfer

```
from : 0x8861…...8be38
to   : 0x42df…...a5c6
value: 100

function _transfer(address from, address to, uint256 value) internal {
   require(value <= _balances[from]);
   require(to != address(0));

   _balances[from] = _balances[from].sub(value);
   _balances[to] = _balances[to].add(value);
   emit Transfer(from, to, value);
 }
```

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	99,900
0x42df……a5c6	100

Approve

- **approve**
  - spender를 지정하여, 자신의 토큰을 사용할 수 있게 허가
  - spender는 transferFrom을 통해 허가된 범위 안에서 토큰을 사용할 수 있음

```
function approve(address spender, uint256 value) public returns (bool) {
   require(spender != address(0));

   _allowed[msg.sender][spender] = value;
   emit Approval(msg.sender, spender, value);
   return true;
 }
```

Approve

```
msg.sender : 0x8861…...8be38
spender    : 0x42df…...a5c6
value      : 100

function approve(address spender, uint256 value) public returns (bool) {
   require(spender != address(0));

   _allowed[msg.sender][spender] = value;
   emit Approval(msg.sender, spender, value);
   return true;
 }
```

_allowed	
<KEY> address	<Value> mapping

Approve

```
msg.sender : 0x8861…...8be38
spender    : 0x42df…...a5c6
value      : 100

function approve(address spender, uint256 value) public returns (bool) {
   require(spender != address(0));

   _allowed[msg.sender][spender] = value;
   emit Approval(msg.sender, spender, value);
   return true;
 }
```

_allowed	
<KEY> address	<Value> mapping
0	0

Approve

```
msg.sender : 0x8861…...8be38
spender    : 0x42df…...a5c6
value      : 100

function approve(address spender, uint256 value) public returns (bool) {
   require(spender != address(0));

   _allowed[msg.sender][spender] = value;
   emit Approval(msg.sender, spender, value);
   return true;
 }
```

_allowed	
<KEY> address	<Value> mapping
0x8861…...8be38	0

Approve

```
msg.sender : 0x8861…...8be38
spender    : 0x42df…...a5c6
value      : 100

function approve(address spender, uint256 value) public returns (bool) {
   require(spender != address(0));

   _allowed[msg.sender][spender] = value;
   emit Approval(msg.sender, spender, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
	<KEY> address	<Value> uint256
0x8861…...8be38	0	0

Approve

```
msg.sender : 0x8861…...8be38
spender    : 0x42df…...a5c6
value      : 100

function approve(address spender, uint256 value) public returns (bool) {
   require(spender != address(0));

   _allowed[msg.sender][spender] = value;
   emit Approval(msg.sender, spender, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
	<KEY> address	<Value> uint256
0x8861…...8be38	0x42df.a5c6	100

Approve

```
msg.sender : 0x8861…...8be38
spender    : 0x42df…...a5c6
value      : 100

function approve(address spender, uint256 value) public returns (bool) {
   require(spender != address(0));

   _allowed[msg.sender][spender] = value;
   emit Approval(msg.sender, spender, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
	<KEY> address	<Value> uint256
0x8861…...8be38	0x42df.a5c6	100

TransferFrom

```
msg.sender : 0x42df…...a5c6
from       : 0x8861…...8be38
to         : 0x33ff…...81ga
value      : 10

function transferFrom( address from, address to, uint256 value ) public
returns (bool) {
   require(value <= _allowed[from][msg.sender]);
   _allowed[from][msg.sender] = _allowed[from][msg.sender].sub(value);
   _transfer(from, to, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
	<KEY> address	<Value> uint256
0x8861…...8be38	0x42df.a5c6	100

TransferFrom

```
msg.sender : 0x42df…...a5c6
from       : 0x8861…...8be38
to         : 0x33ff…...81ga
value      : 10

function transferFrom( address from, address to, uint256 value ) public
returns (bool) {
   require(value <= _allowed[from][msg.sender]);
   _allowed[from][msg.sender] = _allowed[from][msg.sender].sub(value);
   _transfer(from, to, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
	<KEY> address	<Value> uint256
0x8861….. .8be38	0x42df.a5c6	100

TransferFrom

```
msg.sender : 0x42df…...a5c6
from       : 0x8861…...8be38
to         : 0x33ff…...81ga
value      : 10

function transferFrom( address from, address to, uint256 value ) public
returns (bool) {
   require(value <= _allowed[from][msg.sender]);
   _allowed[from][msg.sender] = _allowed[from][msg.sender].sub(value);
   _transfer(from, to, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
	<KEY> address	<Value> uint256
0x8861…...8be38	0x42df.a5c6	90

TransferFrom

```
msg.sender : 0x42df…...a5c6
from       : 0x8861…...8be38
to         : 0x33ff…...81ga
value      : 10

function transferFrom( address from, address to, uint256 value ) public
returns (bool) {
   require(value <= _allowed[from][msg.sender]);
   _allowed[from][msg.sender] = _allowed[from][msg.sender].sub(value);
   _transfer(from, to, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
0x8861…...8be38	<KEY> address	<Value> uint256
	0x42df.a5c6	**90**

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	99,900
0x42df……a5c6	100

TransferFrom

```
msg.sender : 0x42df…...a5c6
from       : 0x8861…...8be38
to         : 0x33ff…...81ga
value      : 10

function transferFrom( address from, address to, uint256 value ) public
returns (bool) {
   require(value <= _allowed[from][msg.sender]);
   _allowed[from][msg.sender] = _allowed[from][msg.sender].sub(value);
   _transfer(from, to, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
0x8861…...8be38	<KEY> address	<Value> uint256
	0x42df.a5c6	**90**

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	99,900
0x42df……a5c6	100

TransferFrom

```
msg.sender : 0x42df…...a5c6
from       : 0x8861…...8be38
to         : 0x33ff…...81ga
value      : 10

function transferFrom( address from, address to, uint256 value ) public
returns (bool) {
   require(value <= _allowed[from][msg.sender]);
   _allowed[from][msg.sender] = _allowed[from][msg.sender].sub(value);
   _transfer(from, to, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
0x8861…...8be38	<KEY> address	<Value> uint256
	0x42df.a5c6	**90**

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	99,890
0x42df……a5c6	100

TransferFrom

```
msg.sender : 0x42df…...a5c6
from       : 0x8861…...8be38
to         : 0x33ff…...81ga
value      : 10

function transferFrom( address from, address to, uint256 value ) public
returns (bool) {
   require(value <= _allowed[from][msg.sender]);
   _allowed[from][msg.sender] = _allowed[from][msg.sender].sub(value);
   _transfer(from, to, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
0x8861…...8be38	<KEY> address	<Value> uint256
	0x42df.a5c6	**90**

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	99,890
0x42df……a5c6	100
0	0

TransferFrom

```
msg.sender : 0x42df…...a5c6
from       : 0x8861…...8be38
to         : 0x33ff…...81ga
value      : 10

function transferFrom( address from, address to, uint256 value ) public
returns (bool) {
   require(value <= _allowed[from][msg.sender]);
   _allowed[from][msg.sender] = _allowed[from][msg.sender].sub(value);
   _transfer(from, to, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
0x8861…...8be38	<KEY> address	<Value> uint256
	0x42df.a5c6	**90**

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	99,890
0x42df……a5c6	100
0x33ff……81ga	10

TransferFrom

```
msg.sender : 0x42df…...a5c6
from       : 0x8861…...8be38
to         : 0x33ff…...81ga
value      : 10

function transferFrom( address from, address to, uint256 value ) public
returns (bool) {
   require(value <= _allowed[from][msg.sender]);
   _allowed[from][msg.sender] = _allowed[from][msg.sender].sub(value);
   _transfer(from, to, value);
   return true;
 }
```

_allowed		
<KEY> address	<Value> mapping	
0x8861…...8be38	<KEY> address	<Value> uint256
	0x42df.a5c6	**90**

_balances	
<KEY> address	<Value> uint256
0x8861……8be38	99,890
0x42df……a5c6	100
0x33ff……81ga	10