OpenZeppelin

# Setting up Access Control

## with Roles

**zpl.in/contracts-workshop**

**Francisco Giordano**
frangio@openzeppelin.com
@frangio_

OpenZeppelin

# Our mission is to protect the open economy

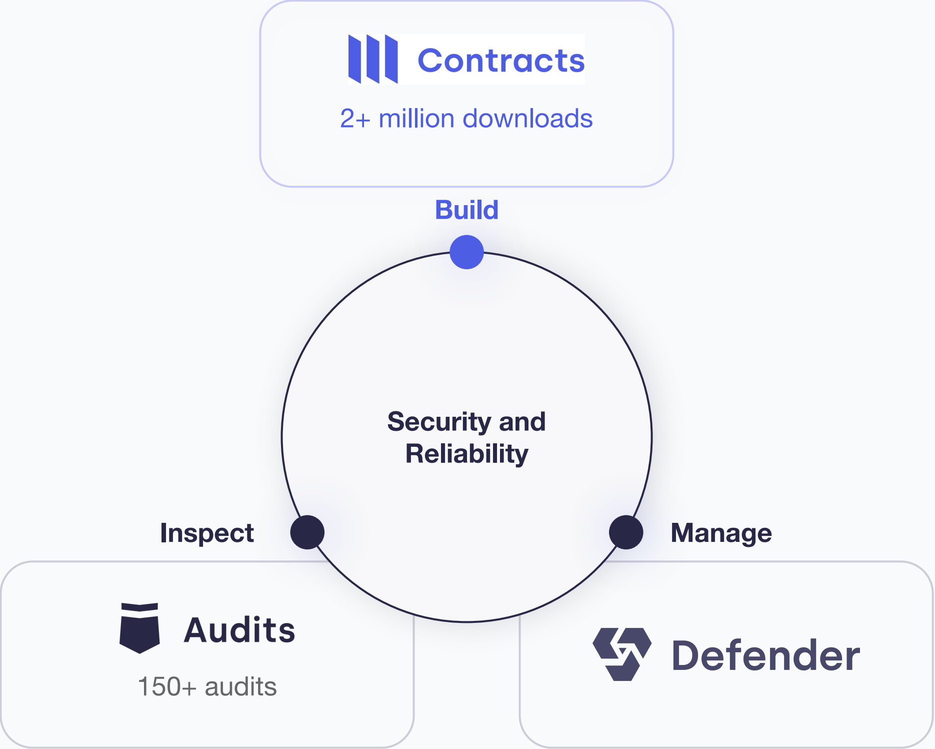
OpenZeppelin is a software company that provides **security audits** and **products** for decentralized systems.

Projects from any size — from new startups to established organizations — trust OpenZeppelin to build, inspect and connect to the open economy.

Compound
coinbase
AAVE
δY/δX
brave
UMA
Balancer
ethereum foundation
augur
celo
CIRCLE
BitGo
pool together
Set
opyn

# Security, Reliability and Risk Management

OpenZeppelin provides a complete suite of **security and reliability products** to build, manage, and inspect all aspects of software development and operations for Ethereum projects.



# OpenZeppelin Contracts

**Token Standards**
ERC20, ERC721, ...

**+ Extensions**
Pausable, snapshots, ...

**Proxies**
Upgradeability, clones

**Security modules**
Reentrancy guard,
pull payments

**Utilities**
Cryptography, math, create2,
data structures, ...

**→ Upgrades Plugins**
Automated upgradeability with
security checks

**Governance**
Timelock, & more coming...

## Access Control

Ownable & Roles

# Ownable

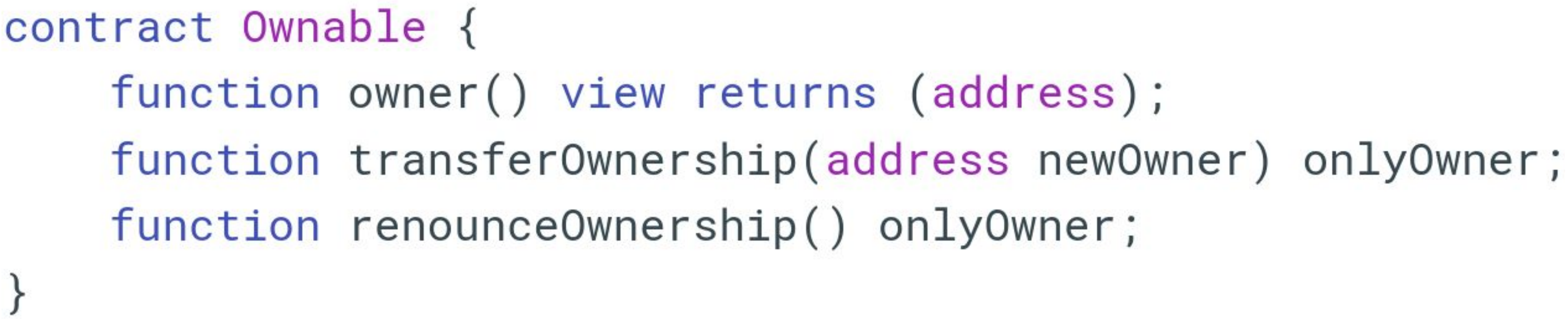
# Simple access control: Ownable

```
NFT

- ownerOf( )
- transfer( )
- mint( )
```

# Simple access control: Ownable

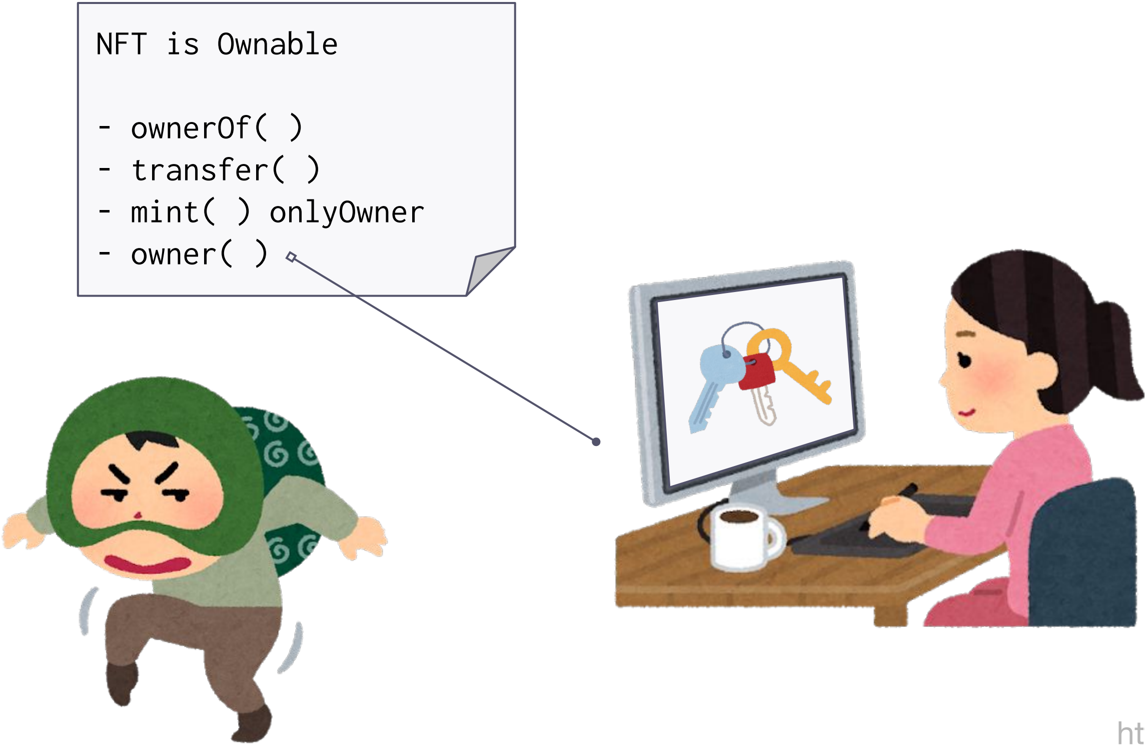
```
NFT is Ownable

- ownerOf( )
- transfer( )
- mint( ) onlyOwner
- owner( )
```

# Ownable Interface



```
contract Ownable {
    function owner() view returns (address);
    function transferOwnership(address newOwner) onlyOwner;
    function renounceOwnership() onlyOwner;
}
```

# Simple access control: Ownable



```
NFT is Ownable

- ownerOf( )
- transfer( )
- mint( ) onlyOwner
- owner( )
```

# Analysing Risks

Assess impact of compromised keys

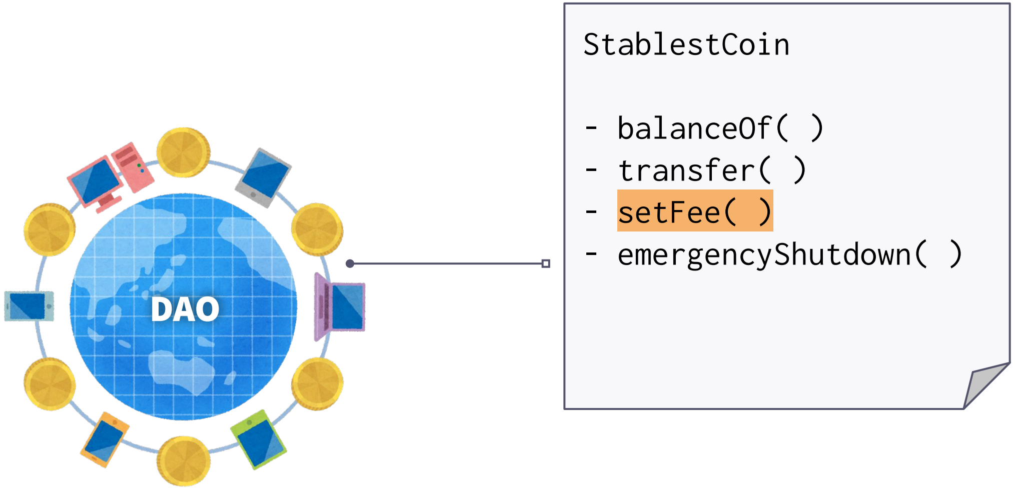
Design key storage system

Plan mitigation measures

Different considerations can apply within a single contract

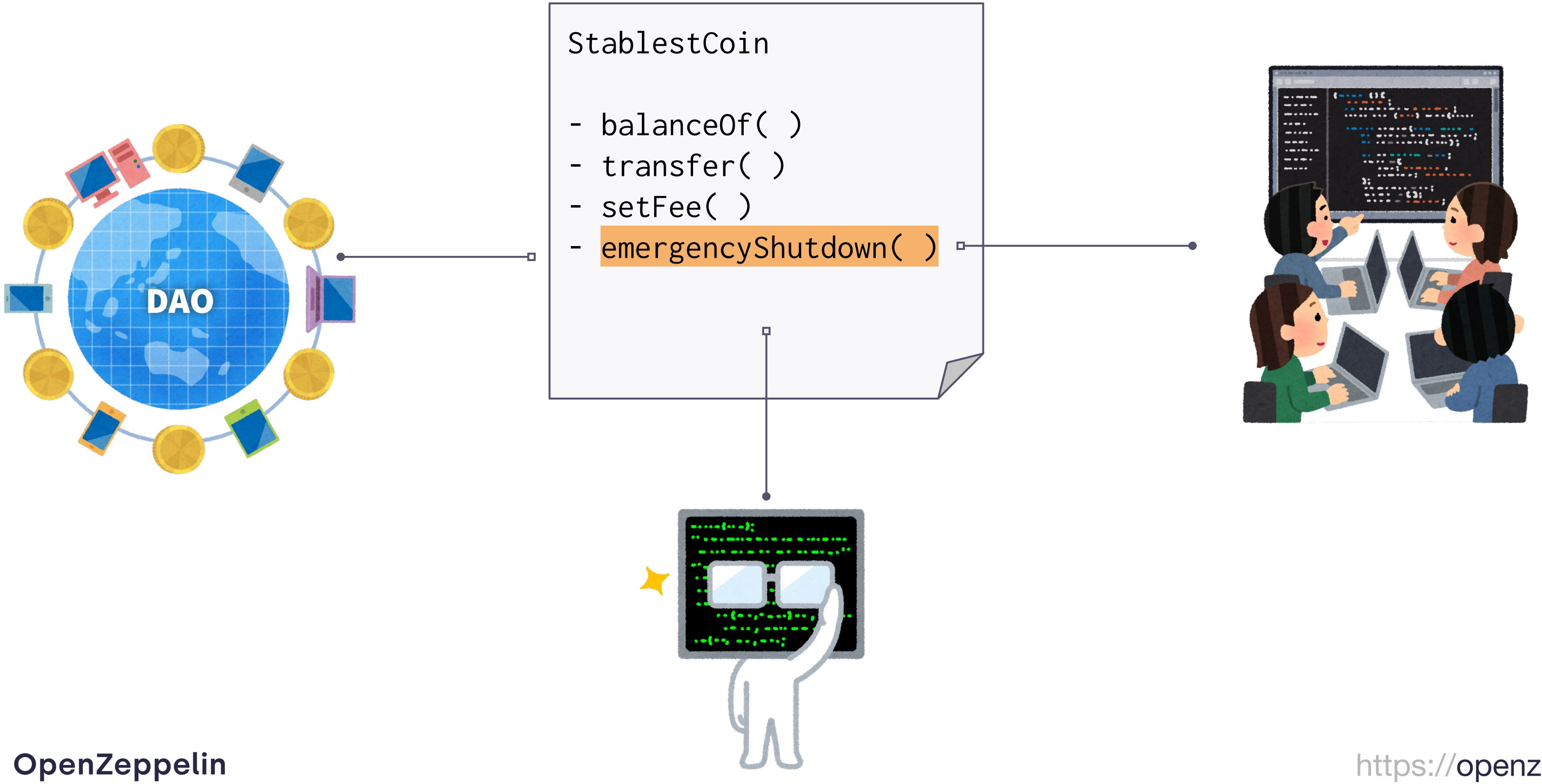
# Granular access control

# Example 2: DeFi Parameters & Emergency Mechanism



```
StablestCoin

- balanceOf( )
- transfer( )
- setFee( )
- emergencyShutdown( )
```

# Example 2: DeFi Parameters & Emergency Mechanism



StablestCoin

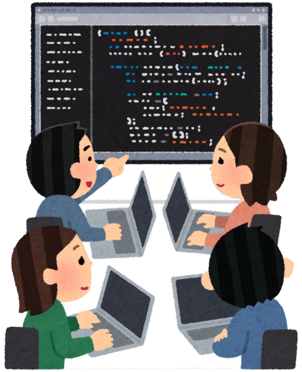
- balanceOf( )
- transfer( )
- setFee( )
- emergencyShutdown( )

# Example 2: DeFi Parameters & Emergency Mechanism

Fee Setter

Responder

```
StablestCoin

- balanceOf( )
- transfer( )
- setFee( )
- emergencyShutdown( )
```



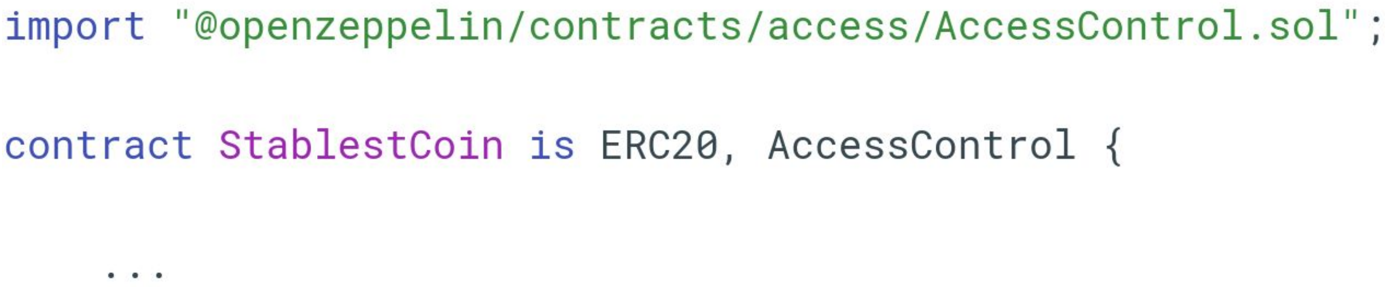
Responder

Responder

# AccessControl

@openzeppelin/contracts/access/AccessControl.sol

# AccessControl



```
import "@openzeppelin/contracts/access/AccessControl.sol";

contract StablestCoin is ERC20, AccessControl {

    ...
```

# AccessControl Interface

```
contract AccessControl {
    function hasRole(bytes32 role, address account) view returns (bool);
    function _setupRole(bytes32 role, address account) internal;
    function grantRole(bytes32 role, address account) onlyAdmin(role);
    function revokeRole(bytes32 role, address account) onlyAdmin(role);
    function getRoleAdmin(bytes32 role) view returns (bytes32);
    function renounceRole(bytes32 role, address account);
}

bytes32 constant MINTER_ROLE = keccak256("MINTER_ROLE");

require(hasRole(MINTER_ROLE, msg.sender));

_setupRole(MINTER_ROLE, msg.sender);
```

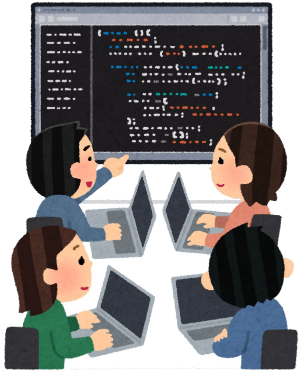
# Granting & Revoking Roles

# Example 2: DeFi Parameters & Emergency Mechanism

Fee Setter

Responder

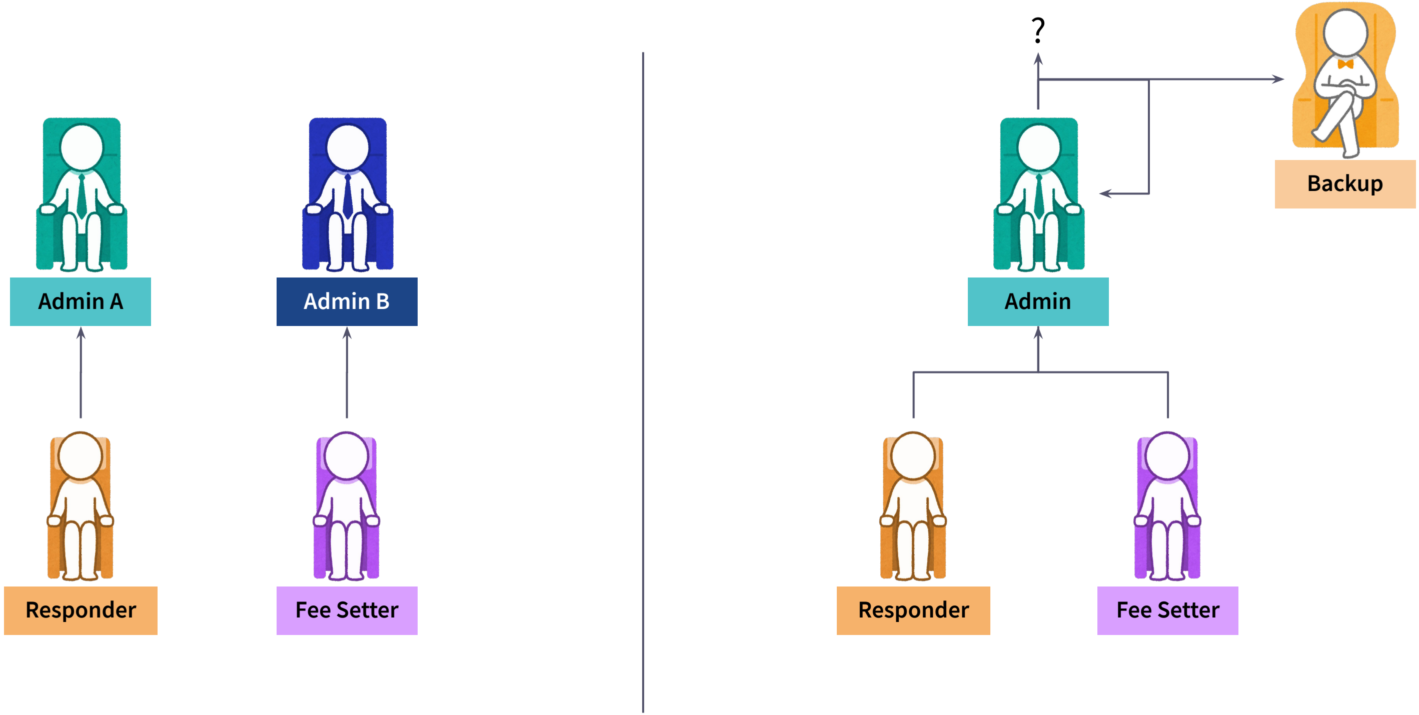
```
StablestCoin

- balanceOf( )
- transfer( )
- setFee( )
- emergencyShutdown( )
```



Responder

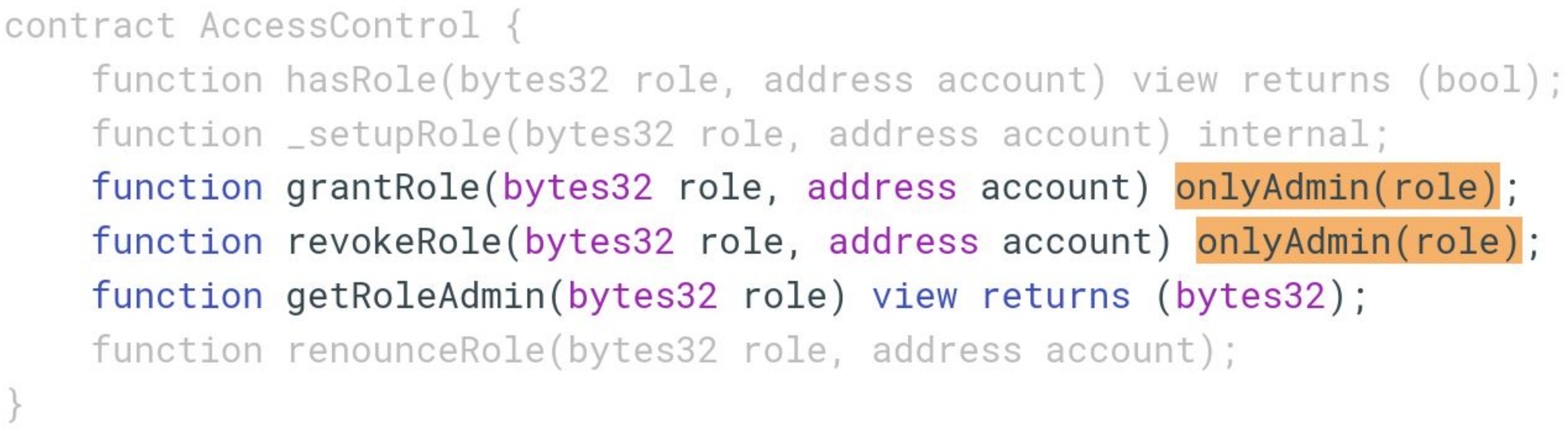
Responder

# Admins


Admin A
Responder
Admin B
Fee Setter
?
Backup
Admin
Responder
Fee Setter

# Admins

Every role has an admin role

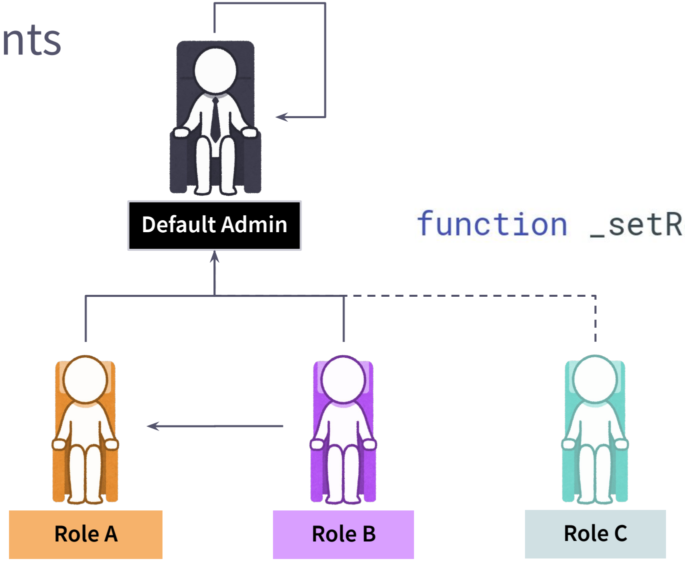


```
contract AccessControl {
    function hasRole(bytes32 role, address account) view returns (bool);
    function _setupRole(bytes32 role, address account) internal;
    function grantRole(bytes32 role, address account) onlyAdmin(role);
    function revokeRole(bytes32 role, address account) onlyAdmin(role);
    function getRoleAdmin(bytes32 role) view returns (bytes32);
    function renounceRole(bytes32 role, address account);
}
```

# Default Admin

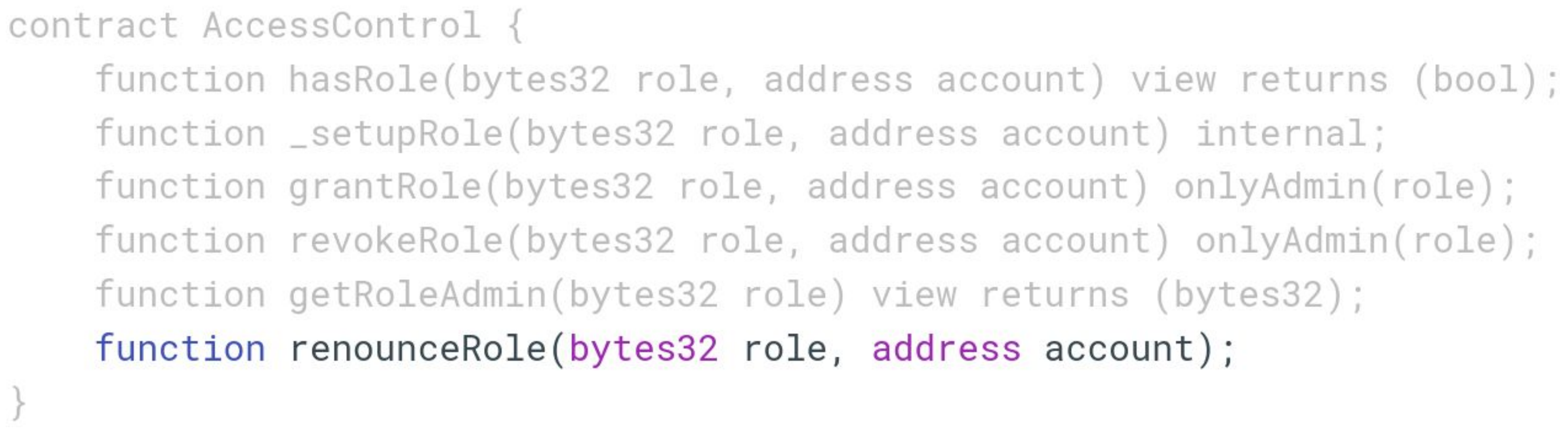
Every role initially has a global default admin role

No default admin accounts



```
function _setRoleAdmin(bytes32 role) internal;
```

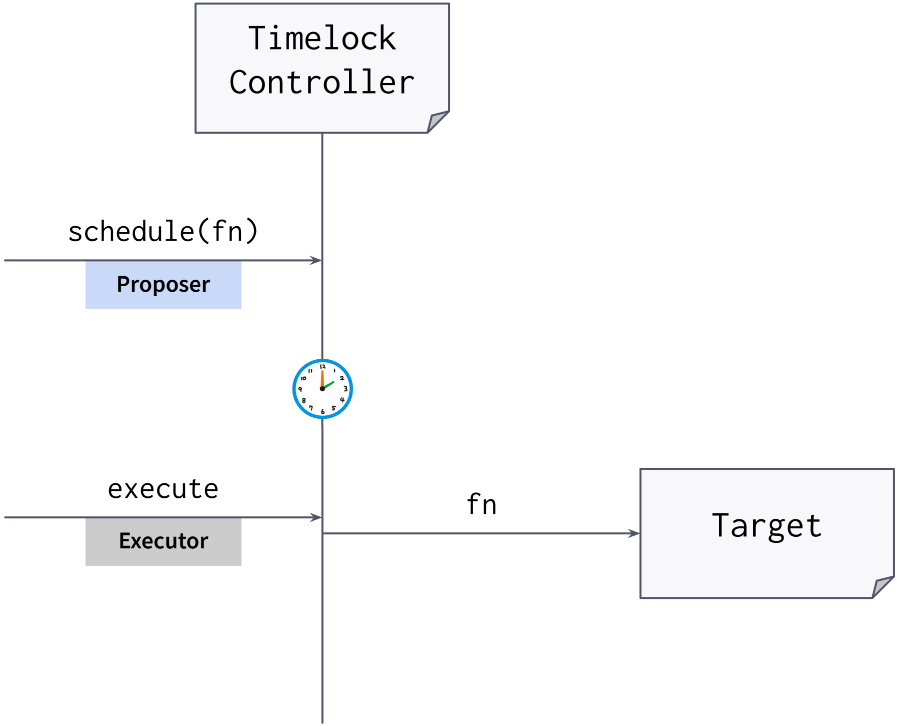
# Renouncing



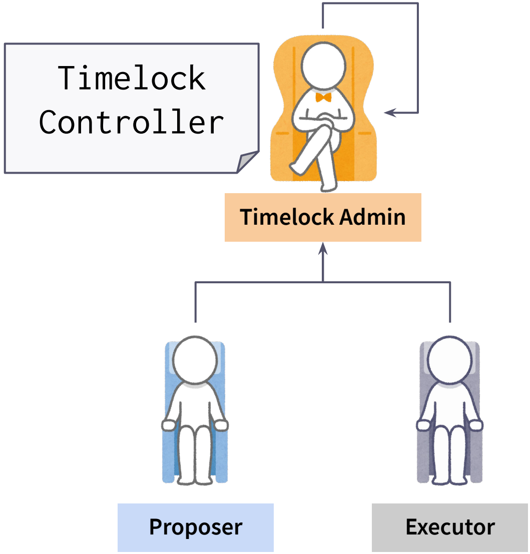
```
contract AccessControl {
    function hasRole(bytes32 role, address account) view returns (bool);
    function _setupRole(bytes32 role, address account) internal;
    function grantRole(bytes32 role, address account) onlyAdmin(role);
    function revokeRole(bytes32 role, address account) onlyAdmin(role);
    function getRoleAdmin(bytes32 role) view returns (bytes32);
    function renounceRole(bytes32 role, address account);
}
```

# Case Study: TimelockController

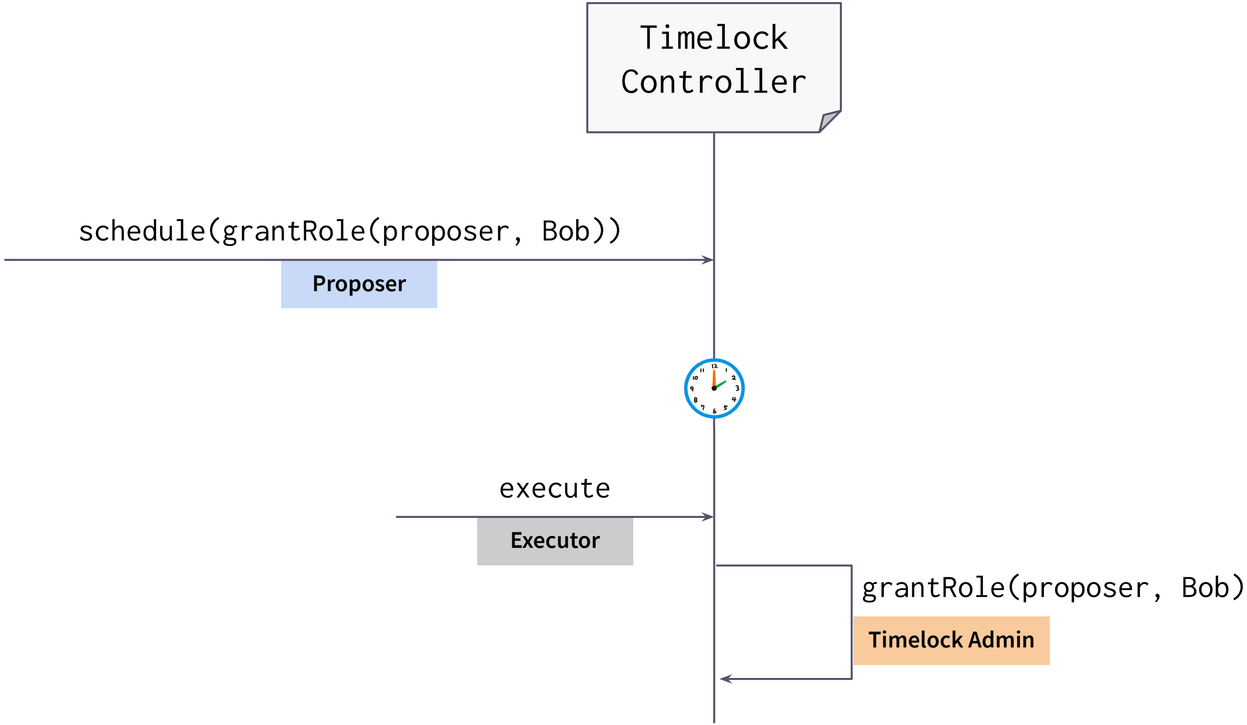
# Case Study: TimelockController



# Case Study: TimelockController



# Case Study: TimelockController



Timelock Controller

schedule(grantRole(proposer, Bob))

Proposer

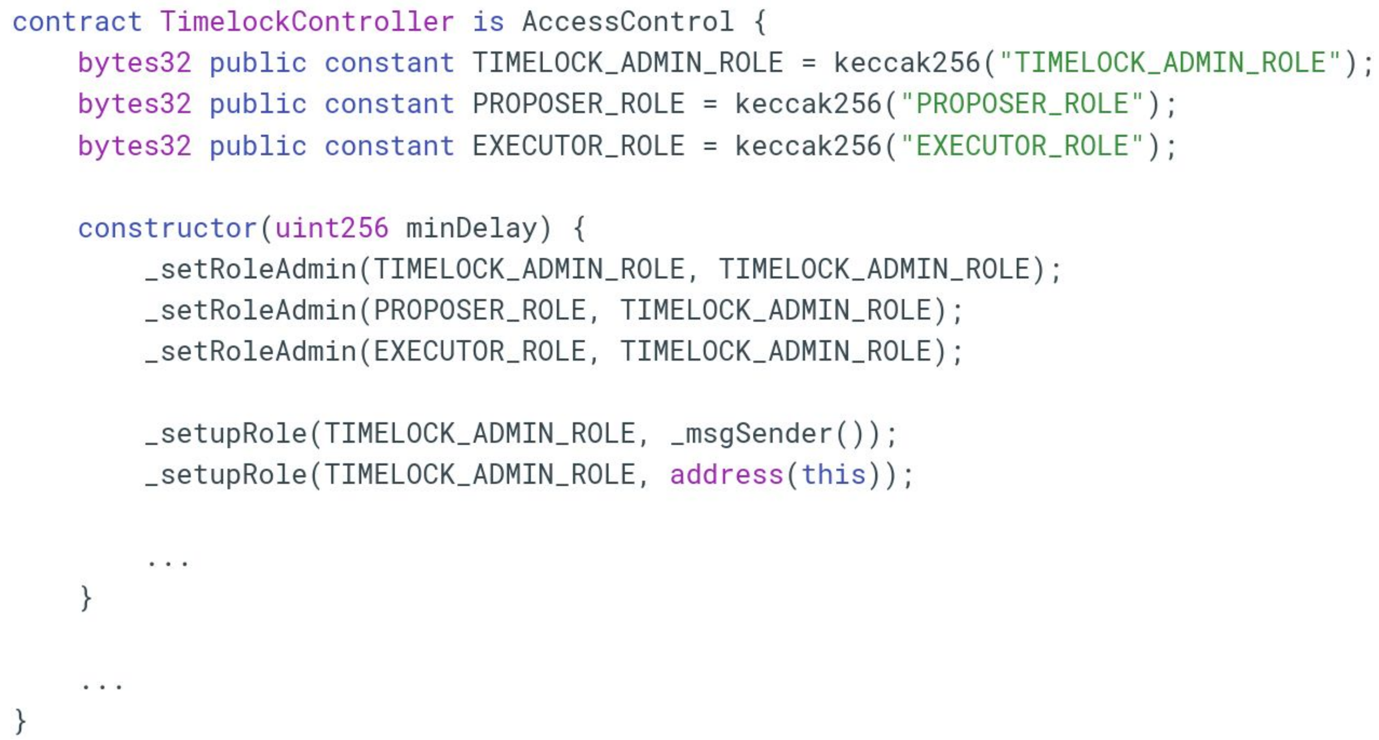
execute

Executor

grantRole(proposer, Bob)

Timelock Admin

# Case Study: TimelockController



```
contract TimelockController is AccessControl {
    bytes32 public constant TIMELOCK_ADMIN_ROLE = keccak256("TIMELOCK_ADMIN_ROLE");
    bytes32 public constant PROPOSER_ROLE = keccak256("PROPOSER_ROLE");
    bytes32 public constant EXECUTOR_ROLE = keccak256("EXECUTOR_ROLE");

    constructor(uint256 minDelay) {
        _setRoleAdmin(TIMELOCK_ADMIN_ROLE, TIMELOCK_ADMIN_ROLE);
        _setRoleAdmin(PROPOSER_ROLE, TIMELOCK_ADMIN_ROLE);
        _setRoleAdmin(EXECUTOR_ROLE, TIMELOCK_ADMIN_ROLE);

        _setupRole(TIMELOCK_ADMIN_ROLE, _msgSender());
        _setupRole(TIMELOCK_ADMIN_ROLE, address(this));

        ...
    }

    ...
}
```

https://zpl.in/wizard

# What’s next? Focus on UX

`onlyRole` modifier

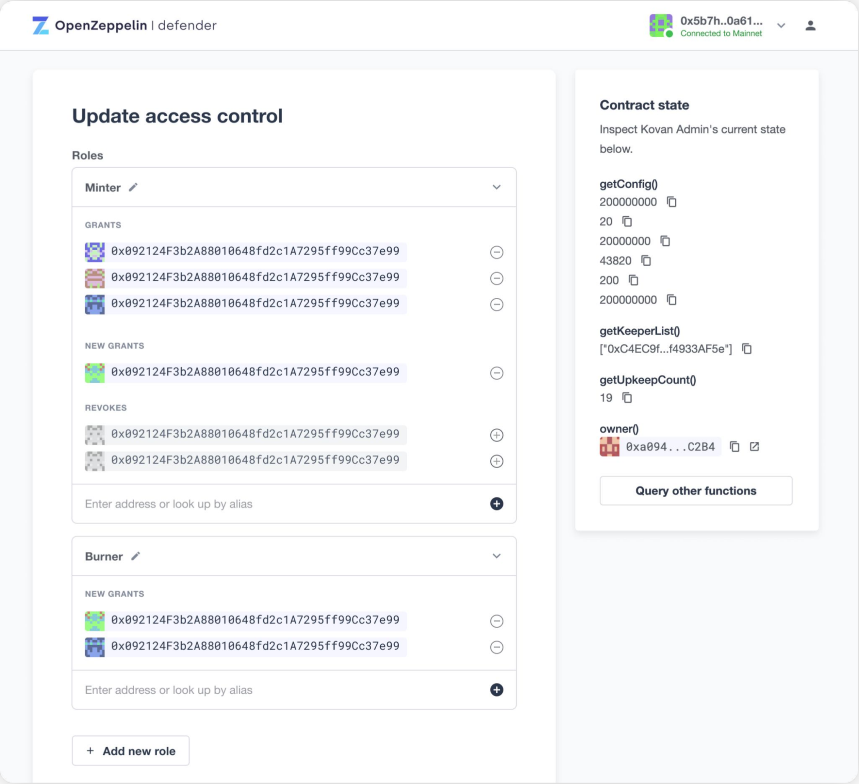
Revert reasons hinting at missing role

Subgraph

`https://zpl.in/subgraph/access-control`

Defender support

`https://defender.openzeppelin.com`



## Update access control

Roles

Minter

GRANTS

- 0x092124F3b2A88010648fd2c1A7295ff99Cc37e99
- 0x092124F3b2A88010648fd2c1A7295ff99Cc37e99
- 0x092124F3b2A88010648fd2c1A7295ff99Cc37e99

NEW GRANTS

- 0x092124F3b2A88010648fd2c1A7295ff99Cc37e99

REVOKES

- 0x092124F3b2A88010648fd2c1A7295ff99Cc37e99
- 0x092124F3b2A88010648fd2c1A7295ff99Cc37e99

Enter address or look up by alias

Burner

NEW GRANTS

- 0x092124F3b2A88010648fd2c1A7295ff99Cc37e99
- 0x092124F3b2A88010648fd2c1A7295ff99Cc37e99

Enter address or look up by alias

## Contract state

Inspect Kovan Admin's current state below.

getConfig()
200000000
20
20000000
43820
200
200000000

getKeeperList()
["0xC4EC9f...f4933AF5e"]

getUpkeepCount()
19

owner()
0xa094...C2B4

**@openzeppelin**/contracts

**docs.**openzeppelin.com

**forum.**openzeppelin.com

**defender.**openzeppelin.com

# Thank you!

## Learn more

openzeppelin.com/**contracts**

**forum**.openzeppelin.com

**docs**.openzeppelin.com

## Contact

🐦 @frangio_

frangio@openzeppelin.com