

VeAlva Smart Contract

Overview

The VeAlva smart contract is an ERC-20 token implementation for the veAlva tokens, designed to provide voting power to Alvara Community. It is an upgradeable token contract that includes burnable functionality. VeAlva tokens are non-transferrable and only collected by the Staking Alva tokens.

Table of Contents

- [Features](#)
- [Contract Details](#)
 - [Contract Name](#)
 - [Inheritance](#)
 - [Initialization](#)
 - [Functions](#)
- [Deployment](#)
- [Directory Structure](#)
- [Testing](#)
- [Contribution](#)

Features

- **ERC-20 Token:** Implements the standard ERC-20 interface for fungible tokens.
- **Upgradeable:** The contract uses OpenZeppelin's upgradeable patterns, allowing for future updates and improvements.
- **Burnable:** Staking Contract can burn veAlva tokens to decay the voting power.
- **Non-Transferrable:** VeAlva token are non-transferrable and only be collected by staking of Alva tokens.
- **Role base access:** Contract implements AccessControl which provide role access on restricted methods.

Contract Details

Contract Name

- **veAlva** (veAlva)

Inheritance

The **veAlva** contract inherits from the following contracts:

- **ERC20Upgradeable:** Provides the standard ERC-20 token functionality in an upgradeable context.
- **Initializable:** Used to initialize contract values which is called on deployment.
- **AccessControlUpgradeable:** It is used to provide AccessControl to the restricted methods.

Initialization

The contract's `initialize` function sets up the token with the following details:

- **Name:** `veAlva`
- **Symbol:** `veAlva`

Functions

- `initialize()`: Initializes the contract, setting the token's name, symbol, and granting default role to deployer.
- `mint(address to, uint256 amount) public onlyRole(ADMIN_ROLE)`: Provide mint functionality to mint given amount of tokens to given recipient address. Only Admin_Role holder address can call this method.
- `burnTokens(address account, uint256 amount) public onlyRole(ADMIN_ROLE)`: Provide burn functionality to burn given amount of tokens to given address. Only Admin_Role holder address can call this method.
- `_beforeTokenTransfer(address from, address to, uint256 amount) internal override`: Overrides the standard `_beforeTokenTransfer` function to restrict the token transfer functionality.

Deployment

To deploy the veAlva smart contract:

1. Ensure your environment is set up with Solidity version `0.8.20` and OpenZeppelin contracts.
2. Compile the contract using your preferred Solidity compiler.
3. Deploy the contract to your desired Ethereum network.
4. Call the `initialize` function to mint the initial supply of tokens.

Directory Structure

```
/contracts
|  └─ tokens
|      └─ veAlva.sol           # The veAlva token contract
```

Testing

To run unit-tests for veAlva tokens, run following command

```
npx hardhat test --grep veAlva
```

Contribution

We welcome contributions from the community to improve and expand the Alvara platform. If you're interested in contributing, please read our [Contribution Guidelines](#) to understand how to get started.