

ERC20 Token Deployment Documentation

Project Overview

This project demonstrates the deployment of a basic ERC20 token smart contract on the Ethereum Sepolia testnet using Foundry as the development toolchain.

Setup and Deployment Process

Prerequisites

- **Foundry** - Ethereum development toolchain
- **MetaMask**
- **Sepolia ETH** - Available from faucets [Alchemy Sepolia Faucet](#)

Step 1: Install Foundry

```
# Install Foundry
curl -L https://foundry.paradigm.xyz | bash
foundryup
```

Step 2: Initialize Project

```
# Create new Foundry project
forge init erc20-token
cd erc20-token

# Install OpenZeppelin contracts
forge install OpenZeppelin/openzeppelin-contracts
```

Step 3: Configure Project

Create `foundry.toml`:

```
[profile.default]
```

```

src = "src"
out = "out"
libs = ["lib"]
remappings = [
    "@openzeppelin/contracts/=lib/openzeppelin-contracts/contracts/"
]

```

Step 4: Create Smart Contract

Create `src/MyToken.sol`:

```

solidity
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.20;

import "@openzeppelin/contracts/token/ERC20/ERC20.sol";
import "@openzeppelin/contracts/access/Ownable.sol";

contract MyToken is ERC20, Ownable {
    constructor(uint256 initialSupply) ERC20("MyToken", "MTK") Ownable(msg.sender) {
        _mint(msg.sender, initialSupply * 10 ** decimals());
    }

    function mint(address to, uint256 amount) public onlyOwner {
        _mint(to, amount);
    }
}

```

Step 5: Deploy to Sepolia

```

# Set environment variables
export
SEPOLIA_RPC_URL="https://eth-sepolia.g.alchemy.com/v2/UJDy0JiEHZ_NGCzmAbt
un"
export PRIVATE_KEY="my_private_key"

# Deploy contract
forge create --rpc-url $SEPOLIA_RPC_URL \
    --private-key $PRIVATE_KEY \
    --constructor-args 1000000 \
    src/MyToken.sol:MyToken

```

Step 6: Verify Contract (Optional)

```
forge verify-contract \  
  --chain-id 11155111 \  
  --compiler-version v0.8.20 \  
  --constructor-args $(cast abi-encode "constructor(uint256)" 1000000) \  
  (CONTRACT_ADDRESS) 0x63d42afE4CECF3BF2009a8c1296bc67521Bd956C  
  
 \  
  src/MyToken.sol:MyToken \  
  --etherscan-api-key AQQM1V1MEWB8DTJ9AIRH4JNP7VC29DQU6B
```