```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract BankAccount {
    address public owner;
    uint256 public balance;

    event Deposit(address indexed account, uint256 amount);
    event Withdrawal(address indexed account, uint256 amount);

    constructor() {
        owner = msg.sender;
    }

    modifier onlyOwner() {
        require(msg.sender == owner, "Only the account owner can call this
function");
        _;
    }

    function deposit() public payable {
        balance += msg.value;
        emit Deposit(msg.sender, msg.value);
    }

    function withdraw(uint256 amount) public onlyOwner {
        require(amount <= balance, "Insufficient funds");
        balance -= amount;
        payable(msg.sender).transfer(amount);
        emit Withdrawal(msg.sender, amount);
    }

    function getBalance() public view returns (uint256) {
        return balance;
    }
}
```