# Web3 Security Researcher:

**Michealking _BuildsWithKing** · 1st

Solidity Smart Contract Developer | Web3 ...

1. Withdraw function fix: Corrected the conditional from if(payment[msg.sender] <= value) to if(value > payment[msg.sender]) and added nonReentrant to prevent over-withdrawal and reentrancy attacks.

2. CollectFee function fix: Changed the check from if(payment[msg.sender] <= 0) to if(amount == 0) to properly revert when there’s nothing to collect.

✕ BUG

Vault.sol

```
function withdraw(uint value) external {
        // ✕ BUG-1
        if (payment[msg.sender] <= value) revert("insufficient balance");
        .
        .
        payable(msg.sender).transfer(value);
}

function collectFee() external onlyOwner {
        // ✕ BUG-2
        if (payment[msg.sender] <= 0) revert("insufficient balance");
        payable(msg.sender).transfer(address(this).balance);
}
```

✓ FIX

Vault.sol

```
function withdraw(uint value) external nonReentrant {
        // ✓ FIX-1
        if (value > payment[msg.sender]) revert insufficientBalance();
        .
        .
        emit withdrawn(value, collect, msg.sender);
}

function collectFee() external onlyOwner {
        uint256 amount = payment[msg.sender];
        // ✓ FIX-2
        if (amount == 0) revert insufficientBalance();
        .
        .
        emit feeCollected(amount, msg.sender);
}
```

🛡 FIX: Selected key fixes from several important suggestions