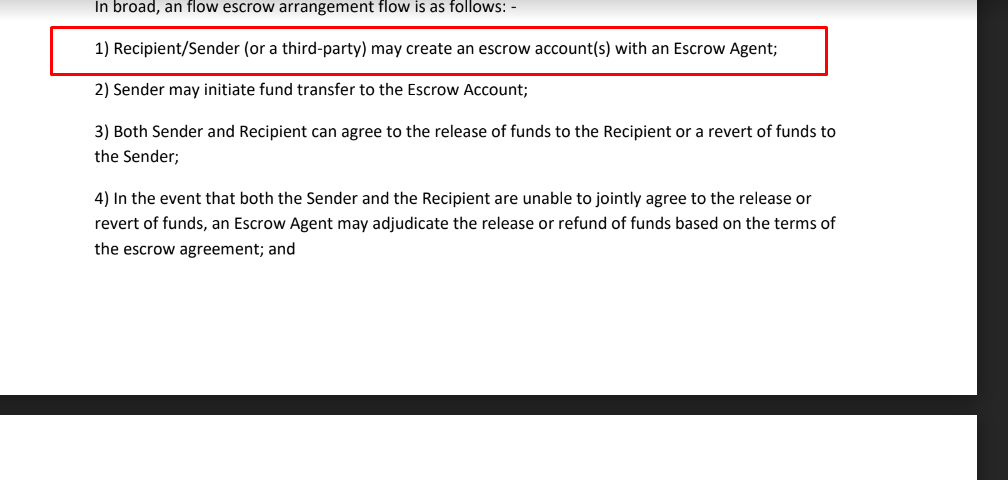
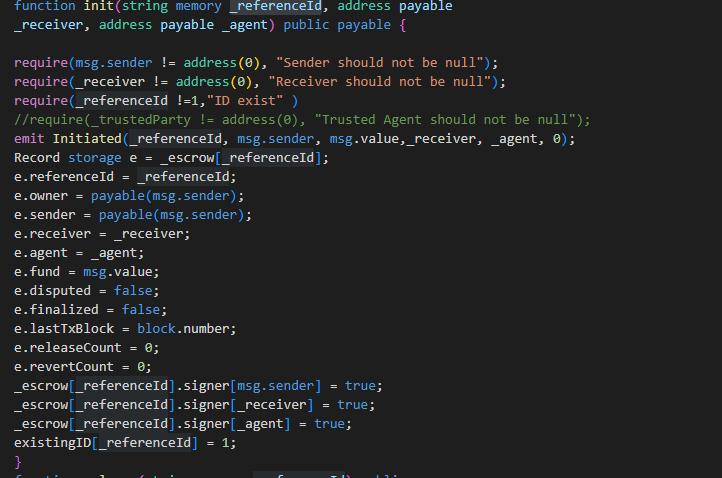
A: code review



This part is missing in the code.

On the code , function init() need to be improved.



According to this init function, anyone can init that referenceID payment. Which means that escrow payment process can be erased by anyone.

It is bug.

There has to be restriction that only referenceID’s sender can call this function.

Improvement

There has to be to create payment propose such as createAccount() function

In the function, there has to be sender , ID.

Like this:

function createAccount(string \_referenceId, address sender) public {

\_escrow[\_referenceId].sender = sender;

existingID[\_referenceId] = 1;

}

function init(string memory \_referenceId, address payable

\_receiver, address payable \_agent) public payable {

require(msg.sender == \_escrow[\_referenceId].sender, "Sender should not be null");

require(\_receiver != address(0), "Receiver should not be null");

require(\_referenceId !=1,"ID exist" )

//require(\_trustedParty != address(0), "Trusted Agent should not be null");

emit Initiated(\_referenceId, msg.sender, msg.value,\_receiver, \_agent, 0);

Record storage e = \_escrow[\_referenceId];

e.referenceId = \_referenceId;

e.owner = payable(msg.sender);

e.sender = payable(msg.sender);

e.receiver = \_receiver;

e.agent = \_agent;

e.fund = msg.value;

e.disputed = false;

e.finalized = false;

e.lastTxBlock = block.number;

e.releaseCount = 0;

e.revertCount = 0;

\_escrow[\_referenceId].signer[msg.sender] = true;

\_escrow[\_referenceId].signer[\_receiver] = true;

\_escrow[\_referenceId].signer[\_agent] = true;

}

So that only sender can initiate that payment.

Or without createAccount function , init function will do all of that. Like this.

function init(string memory \_referenceId, address payable

\_receiver, address payable \_agent) public payable {

require(msg.sender != address(0), "Sender should not be null");

require(\_receiver != address(0), "Receiver should not be null");

require(\_referenceId !=1,"ID exist" )

//require(\_trustedParty != address(0), "Trusted Agent should not be null");

emit Initiated(\_referenceId, msg.sender, msg.value,\_receiver, \_agent, 0);

Record storage e = \_escrow[\_referenceId];

e.referenceId = \_referenceId;

e.owner = payable(msg.sender);

e.sender = payable(msg.sender);

e.receiver = \_receiver;

e.agent = \_agent;

e.fund = msg.value;

e.disputed = false;

e.finalized = false;

e.lastTxBlock = block.number;

e.releaseCount = 0;

e.revertCount = 0;

\_escrow[\_referenceId].signer[msg.sender] = true;

\_escrow[\_referenceId].signer[\_receiver] = true;

\_escrow[\_referenceId].signer[\_agent] = true;

existingID[\_referenceId] = 1;

}

In here, payment Account(proposal) can be created by only sender not receiver or agent.

And if the receiver sign to the escrow(ID) before the sender initialize that payment,

The signature of receiver will be erased by inisitalizing by sender.

So there has to be restrict that receiver only can sign to the proposal only after sender’s initializing.