import algosdk from 'algosdk';

const ALGOD\_API\_ADDR = "https://testnet-api.algonode.cloud";

const ALGOD\_API\_TOKEN = "aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa";

const srcAccountMnemonic = "cricket accuse punch auction border soccer enter weird magic engine possible brown auction enter brave robust dutch daughter energy deer brand kind science above forum";

const receiverAccountMnemonic = "mule shock blur finish drive unveil yard citizen phrase advice saddle mirror laptop huge object seed pizza inch inmate open legend effort laugh able purse";

const srcAccount = algosdk.mnemonicToSecretKey(srcAccountMnemonic);

const receiverAccount = algosdk.mnemonicToSecretKey(receiverAccountMnemonic);

const algodClient = new algosdk.Algodv2(ALGOD\_API\_TOKEN, ALGOD\_API\_ADDR, '');

async function main() {

const transParams = await algodClient.getTransactionParams().do();

const amount = algosdk.algosToMicroalgos(0.201);

console.log(amount.toString());

const tx1 = algosdk.makePaymentTxnWithSuggestedParamsFromObject({

from: srcAccount.addr,

to: receiverAccount.addr,

amount: amount,

note: new Uint8Array(Buffer.from("pay message")),

suggestedParams: transParams

});

const assetOptInTx = algosdk.makeAssetTransferTxnWithSuggestedParamsFromObject({

from: receiverAccount.addr,

to: receiverAccount.addr,

assetIndex: 665357197,

amount: 0,

suggestedParams: transParams

});

const assetTransferTx = algosdk.makeAssetTransferTxnWithSuggestedParamsFromObject({

from: srcAccount.addr,

to: receiverAccount.addr,

assetIndex: 665357197,

amount: 1,

suggestedParams: transParams

});

const txns = [tx1, assetOptInTx, assetTransferTx];

const groupID = algosdk.assignGroupID(txns);

const signedTx1 = tx1.signTxn(srcAccount.sk);

const signedTx2 = assetOptInTx.signTxn(receiverAccount.sk);

const signedTx3 = assetTransferTx.signTxn(srcAccount.sk);

const signedTxGroup = [signedTx1, signedTx2, signedTx3];

try {

const response = await algodClient.sendRawTransaction(signedTxGroup).do();

const confirmedTxn = await algosdk.waitForConfirmation(algodClient, response.txId, 4);

console.log(`Transaction ID: ${response.txId}\nConfirmed round: ${confirmedTxn['confirmed-round']}`);

} catch (e) {

console.error(e.message);

}

}

main().catch(console.error);