



XRP Ledger

Swiss Hacks



Marco Neri

XRPL Developer Relations
Ripple

XRPL Transactions





```
const xrpl = require("xrpl");
```

```
async function main() {  
  // 1. Connect  
  // 2. Send Message to XRPL with SubmitAndAwait  
  // 3. Disconnect  
}
```



Connect

```
const client = new  
xrpl.Client("wss://s.altnet.ripple.net:51233");  
await client.connect();
```



Prepare a message

```
const prepared = await client.autofill({  
  "TransactionType": "Payment",  
  "Account": "rwUovZSYAD6DtgAmMR3VxC8TtZAcHrJPLm",  
  "Amount": xrp1.xrpToDrops("10"),  
  "Destination": "rQDQytvKuTQ4MrvCBmo5f1WnZkV1dQuwox" });
```



```
SubmitAndWait
```

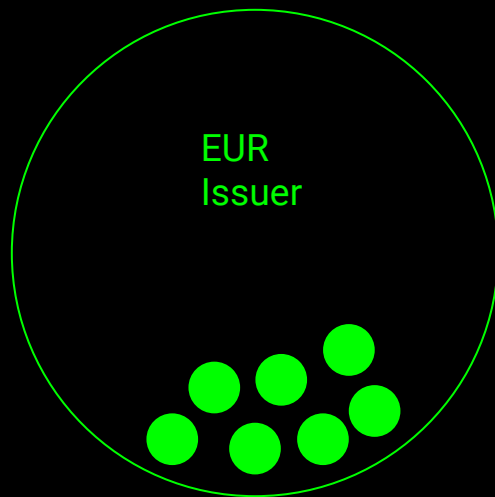
```
const signed = wallet.sign(prepared)
```

```
// Submit signed blob
```

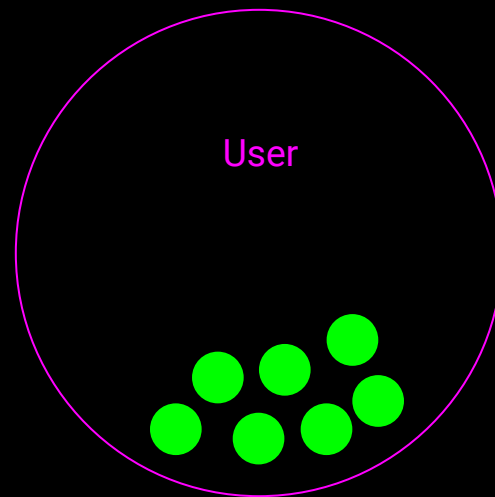
```
try {
```

```
const submit_result = await client.submitAndWait(signed.tx_blob)
```

Token Issuance



Token Issuance



Connect to websocket

```
const client = new xrpl.Client("wss://s.devnet.rippletest.net:51233/");  
await client.connect();
```



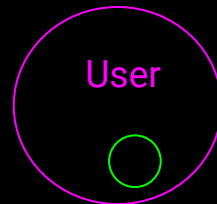
Enable clawback for issuer

```
let transaction = await client.submitAndWait(  
  {  
    TransactionType: "AccountSet",  
    Account: issuer.address,  
    SetFlag: xrpl.AccountSetAsfFlags.asfAllowTrustLineClawback,  
  },  
  { autofill: true, wallet: issuer }  
);
```



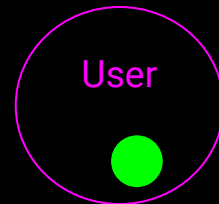
Trusting the issuer

```
transaction = await client.submitAndWait(  
  {  
    TransactionType: "TrustSet",  
    Account: holder.classicAddress,  
    LimitAmount: {  
      currency: "EUR",  
      issuer: issuer.classicAddress,  
      value: "1000",  
    },  
  },  
  { autofill: true, wallet: holder }  
);  
  
{x}
```



Transfer funds to the holder

```
transaction = await client.submitAndWait(  
  {  
    TransactionType: "Payment",  
    Account: issuer.classicAddress,  
    Destination: holder.classicAddress,  
    Amount: {  
      currency: "EUR",  
      issuer: issuer.classicAddress,  
      value: "1000",  
    },  
  },  
  { autofill: true, wallet: issuer }  
);  
{×}
```



Clawback from the holder

```
transaction = await client.submitAndWait(  
  {  
    TransactionType: "Clawback",  
    Account: issuer.classicAddress,  
    Amount: {  
      currency: "EUR",  
      issuer: holder.classicAddress,  
      value: "700",  
    },  
  },  
  { autofill: true, wallet: issuer }  
);
```

{X}

