A Permission Sharing Blockchain Network using Hyperledger Fabric and Composer

We will set up a trading files network. Peers will be able to trade cards among themselves.

Hyperledger Fabric is an open source framework for making private (permissioned) blockchain business networks, where identities and roles of members are known to other members. The network built on fabric serves as the back-end, with a client-side application front-end. SDK's are available for Nodejs and Java to build client applications, with Python and Golang support coming soon.

Hyperledger Composer is a set of Javascript based tools and scripts which simplify the creation of Hyperledger Fabric networks. Using these tools, we can generate a business network archive (BNA) for our network. Composer broadly covers these components:

Business Network Archive (BNA): Composer allows us to package a few different files and generate an archive which can then be deployed onto a Fabric network. To generate this archive, we need:

> **Network Model** — A definition of the resources present in the network. These resources include Assets, Participants, and Transactions. We will come back to these later.
>
> **Business Logic** — Logic for the transaction functions
>
> **Access Control Limitations** — Contains various rules which define the rights of different participants in the network. This includes, but is not limited to, defining what Assets the Participants can control.

Composer Playground: A web based user interface that we can use to model and test our business network. Playground is good for modelling simple Proofs of Concept, as it uses the browser's local storage to simulate the blockchain network. However, if we are running a local Fabric runtime and have deployed a network to it, we can also access that using Playground. In this case, Playground isn't simulating the network, it's communicating with the local Fabric runtime directly.

Prerequisites

Memory Requirement: 20GB (minimum)

Docker Engine and Docker Compose
Nodejs and NPM
Git
Python 2.7.x

For Ubuntu users, Hyperledger has a bash script available to make this process extremely easy. Run the following commands in your terminal:

```
curl -O https://hyperledger.github.io/composer/latest/prereqs-ubuntu.sh
chmod u+x prereqs-ubuntu.sh
./prereqs-ubuntu.sh
```

Run the following commands in your Terminal, and make sure you're **NOT** using sudo when running npm commands.

```
npm install -g composer-cli
npm install -g composer-rest-server
npm install -g composer-playground
npm install -g yo generator-hyperledger-composer
```

Log off and log in again.

Installing a local Hyperledger Fabric runtime

Unzip fabric-dev-serves.zip
Go to *fabric-dev-serves/*

```
export FABRIC_VERSION=hlfv12
./downloadFabric.sh
./startFabric.sh
./createPeerAdminCard.sh
```

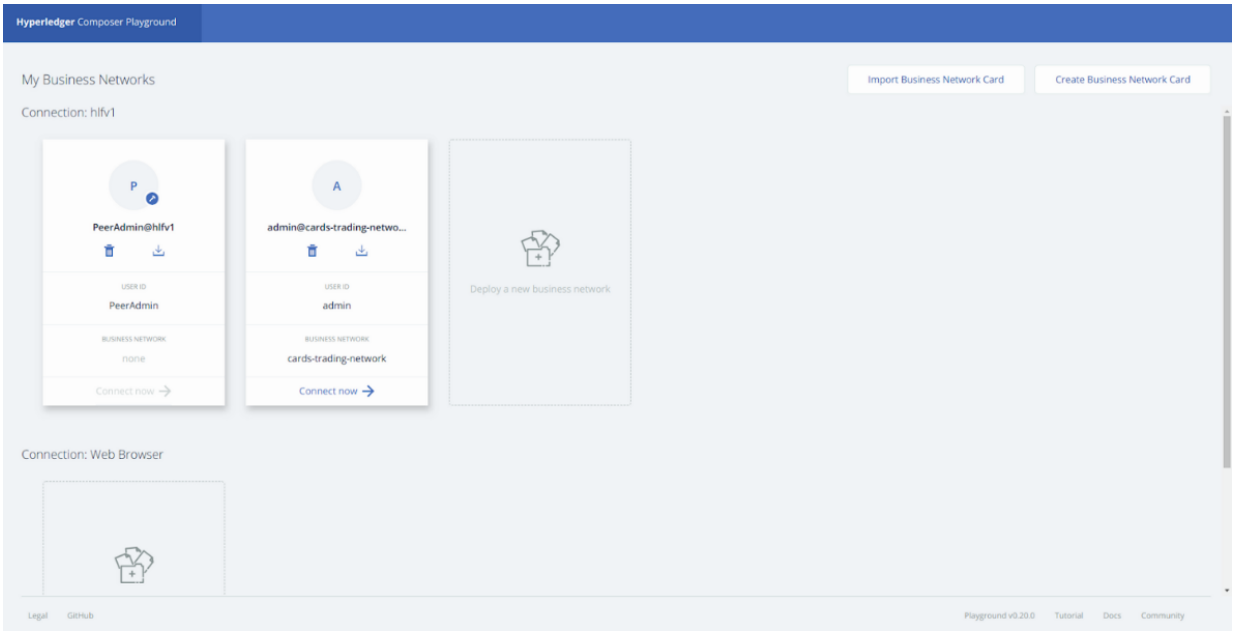
Finally, we generate a PeerAdmin card. Participants in a Fabric network can have business network cards. The holder of the PeerAdmin business card has the authority to deploy, delete, and manage business networks on this Fabric runtime.

Importing a Business Network Archive (BNA)

Go to the project Directory. In this case *fabric-dev-serves/file-trading-network.* Run below commands,

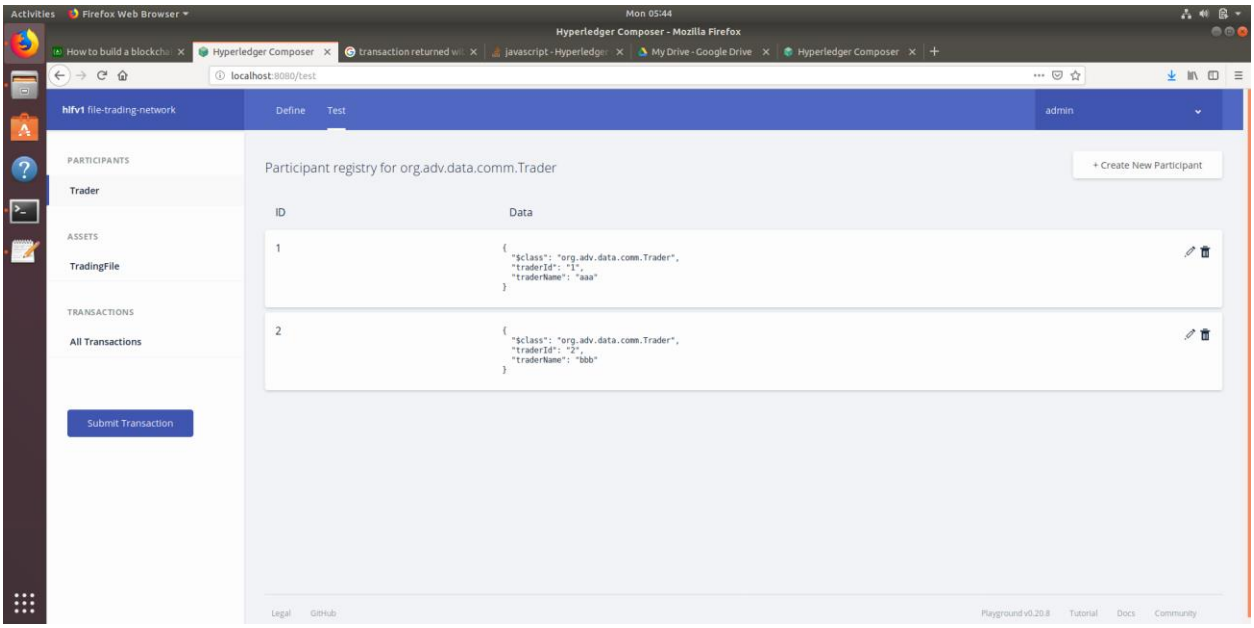
```
composer archive create --sourceType dir --sourceName .
composer network install --archiveFile file-trading-network@0.0.3.bna --card PeerAdmin@hlfv1

composer network start --networkName file-trading-network –networkVersion 0.0.3 --networkAdmin admin --networkAdminEnrollSecret adminpw –card PeerAdmin@hlfv1 --file file-trading-admin.card

composer card import --file file-trading-admin.card
composer network ping --card admin@file-trading-network
```

Now that our network is up and running on Fabric, we can start Composer Playground to interact with it. To do this, type `composer-playground` in Terminal and open up *http://localhost:8080/* in your browser and you should see something similar to this:

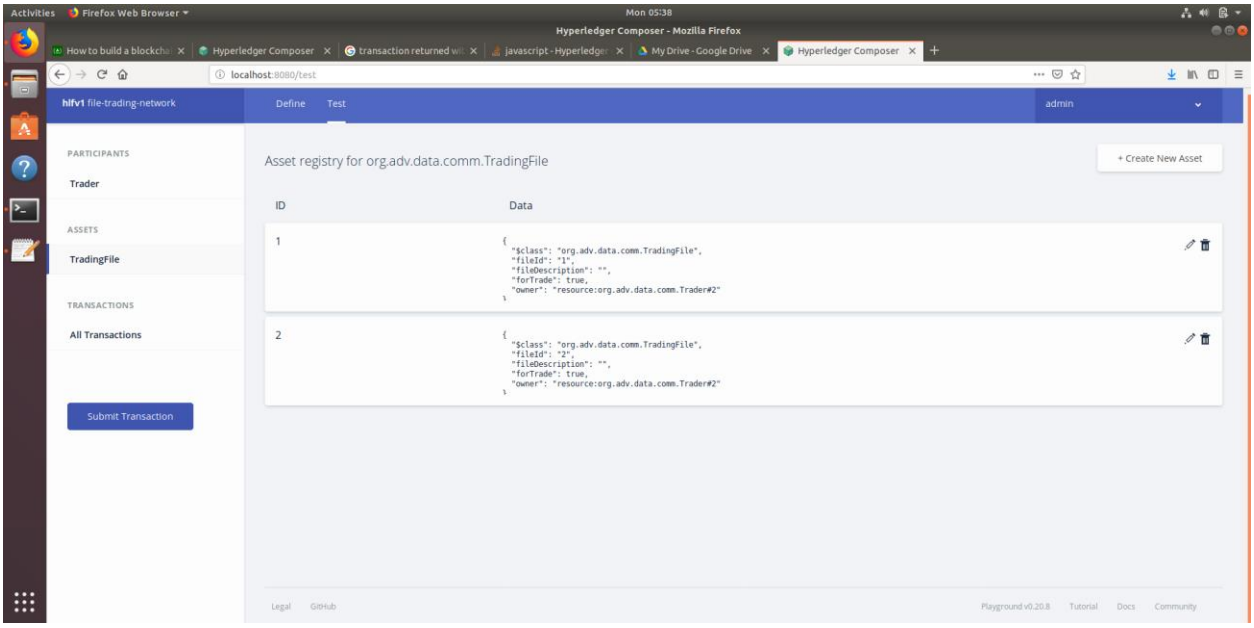


Click Connect now on the admin@file-trading-network.

Go to Test > Trader > Create New Participant > to add traders

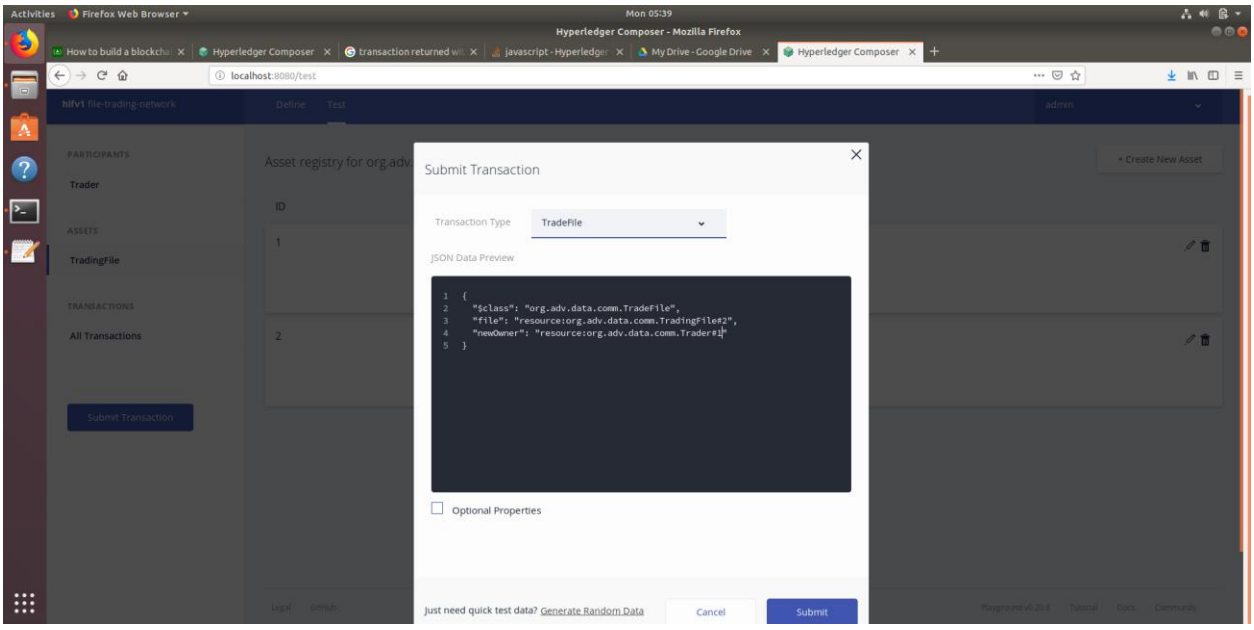


Now, let's make some Assets. Click on TradingFile > Create New Asset.



While creating new asset for some user, please put *forTrade = true*

Click on Submit Transaction to transfer give Permission to new owner of your shared file.



Go to All Transaction tab to see all the File Transactions that has been done.