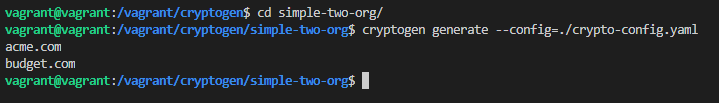
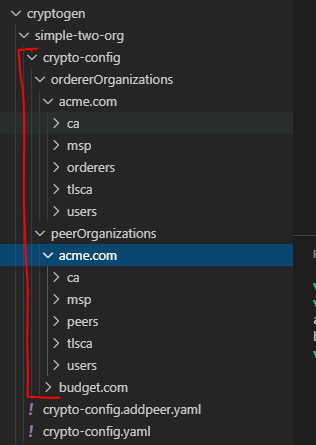
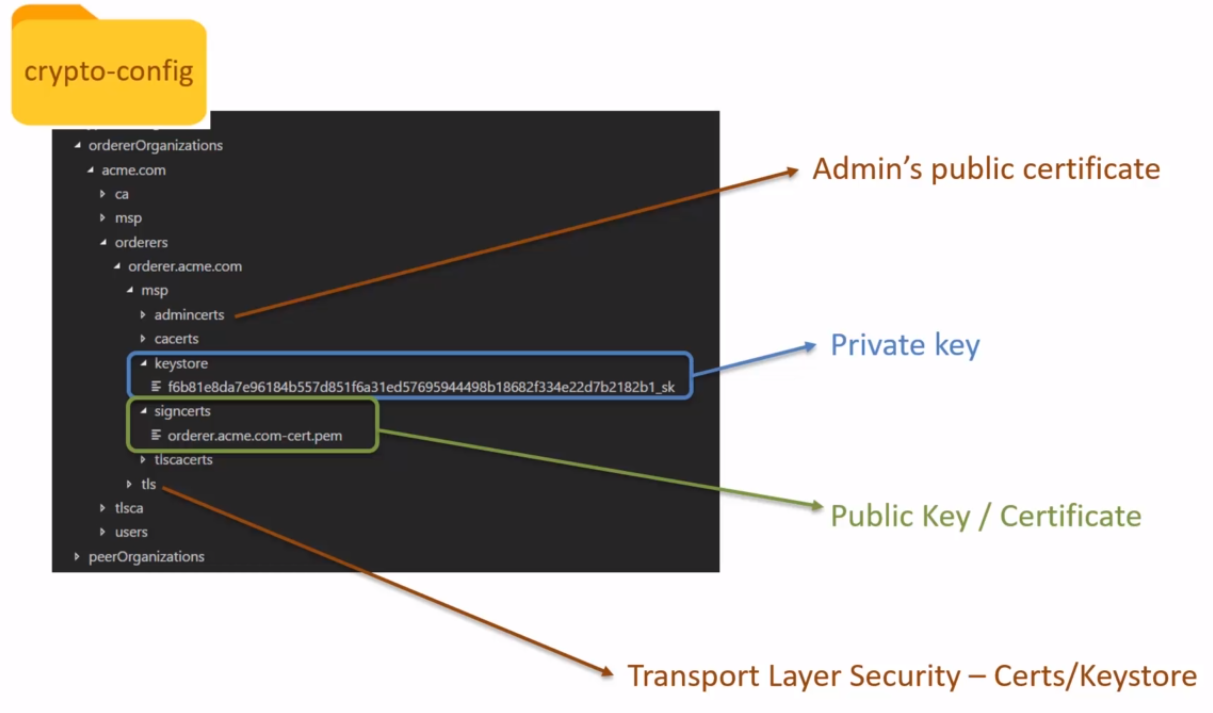
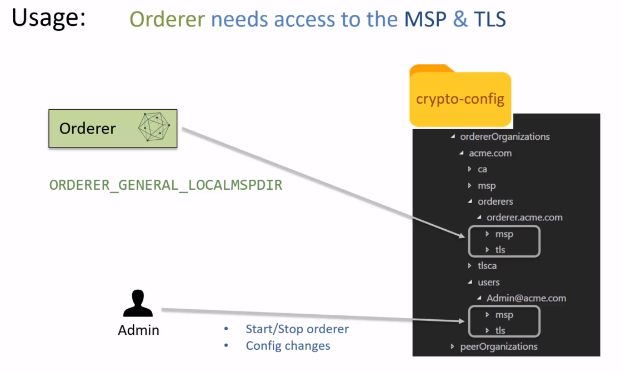
**Cryptogen Tool**

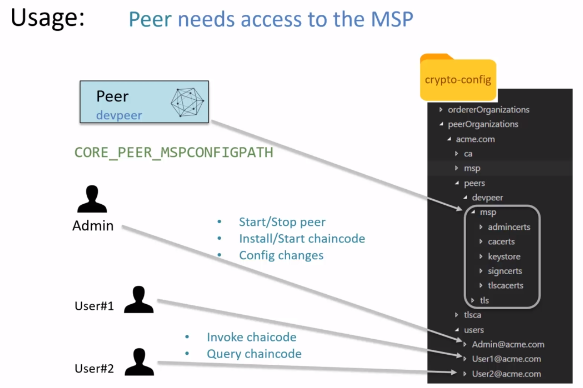


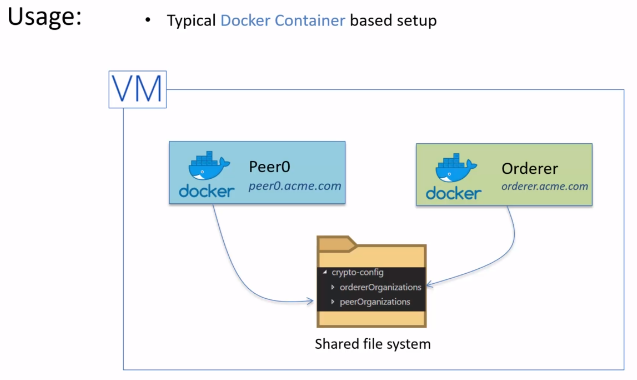
Created this folder

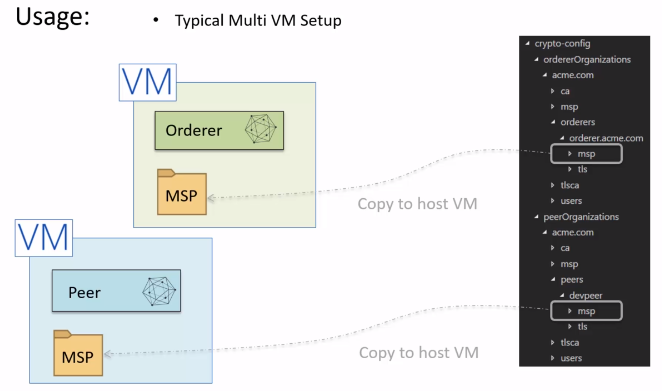


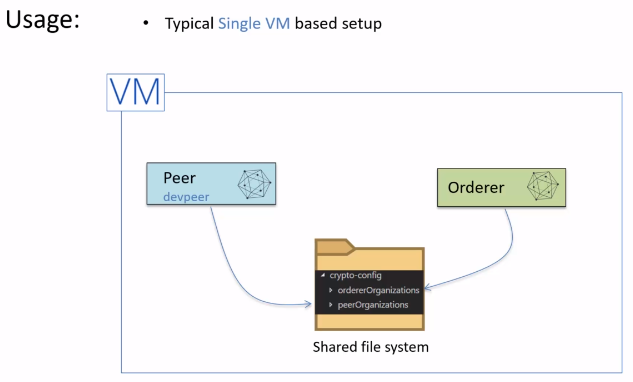








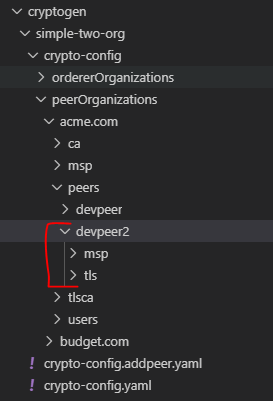


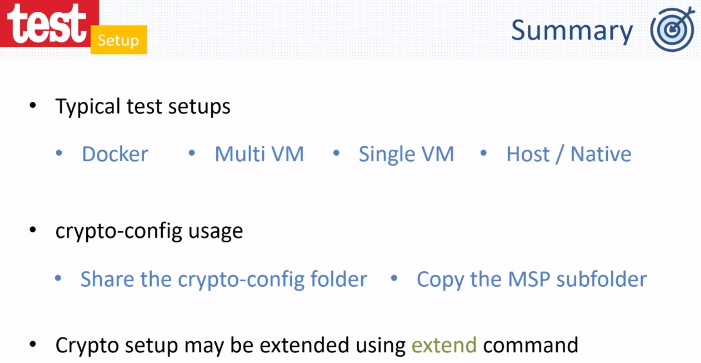


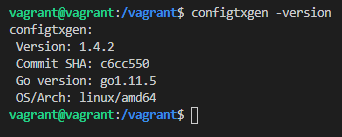
Here containers are not in use.

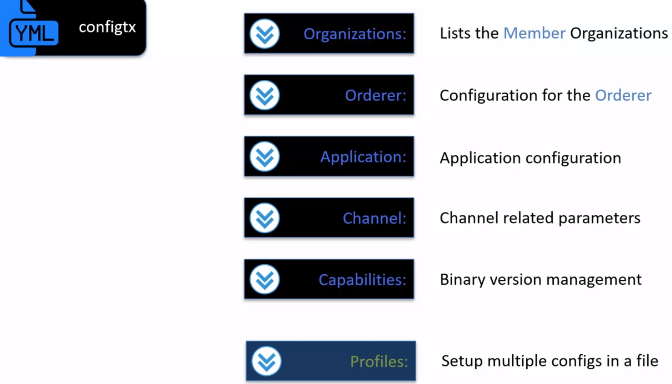


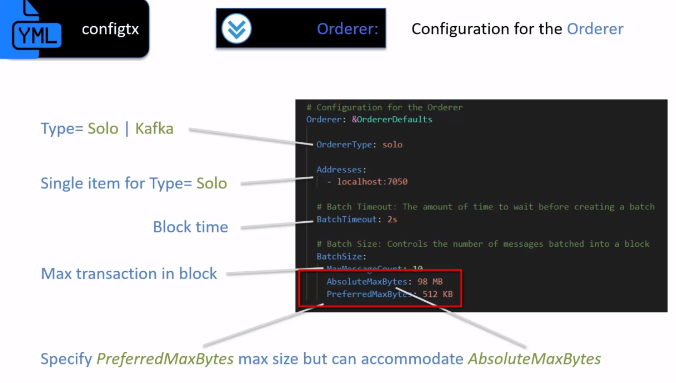


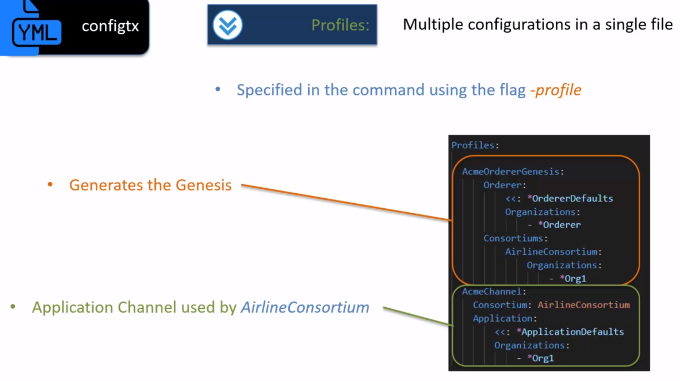




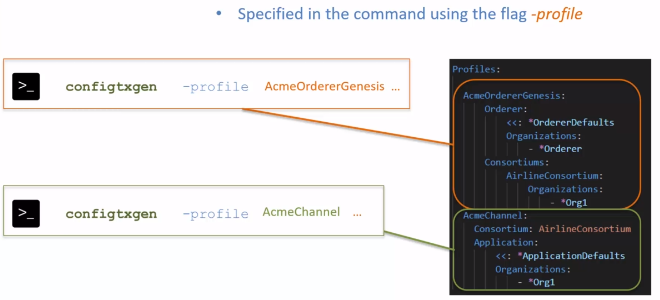




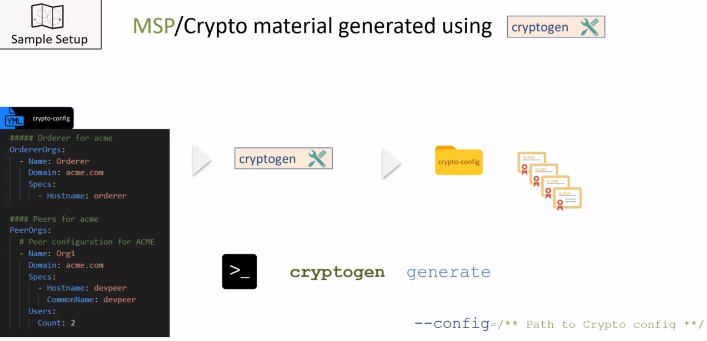




How to access these profiles



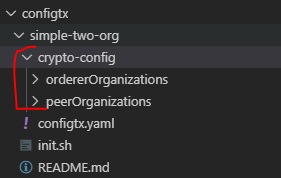


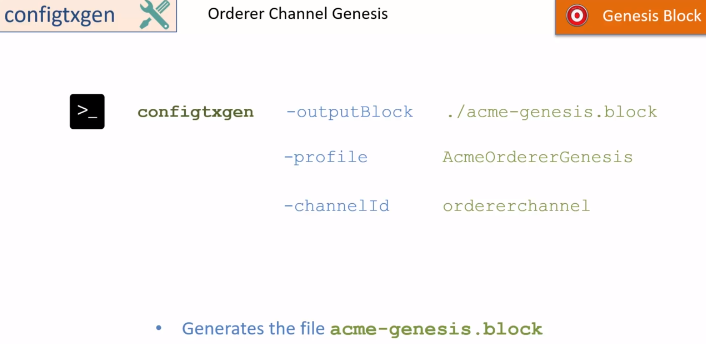


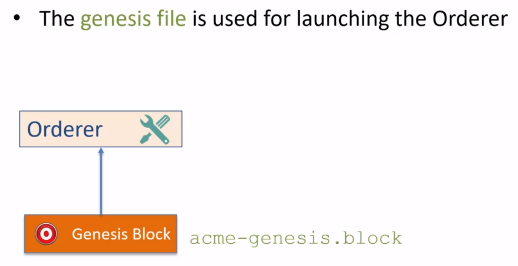
cryptogen generate --config=../../cryptogen/simple-two-org/crypto-config.yaml

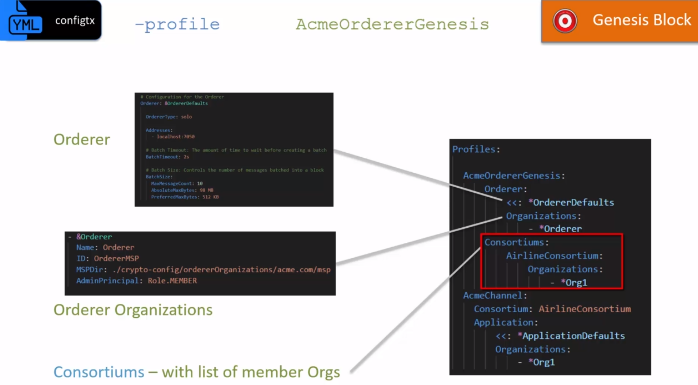


It created below folder

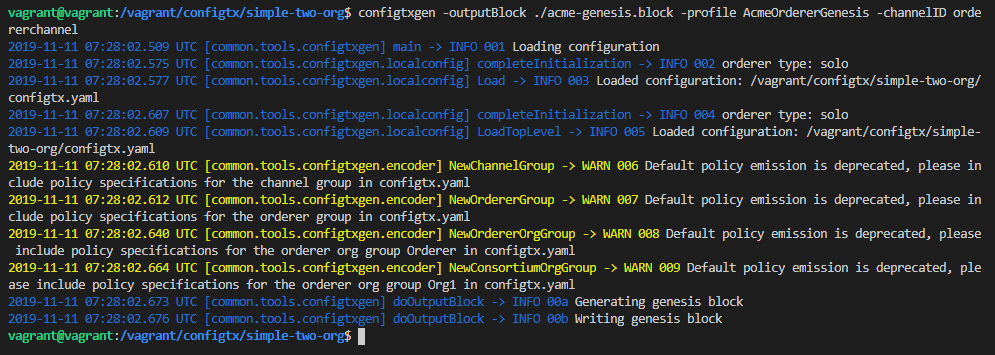




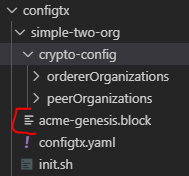




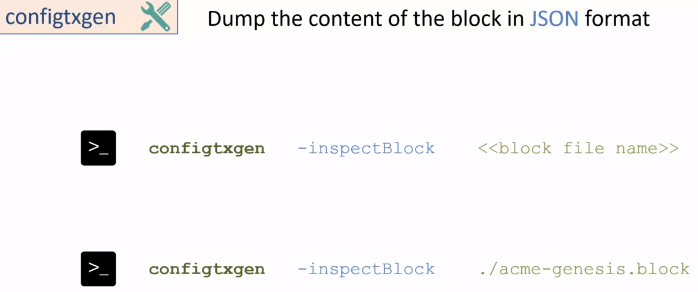
configtxgen -outputBlock ./acme-genesis.block -profile AcmeOrdererGenesis -channelID ordererchannel



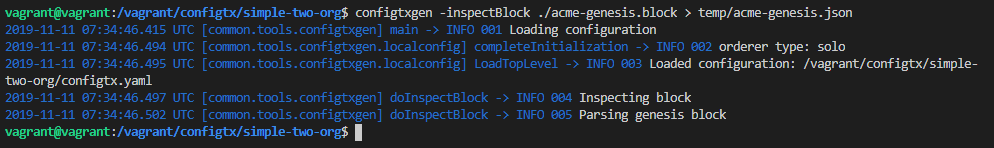
New file is generated

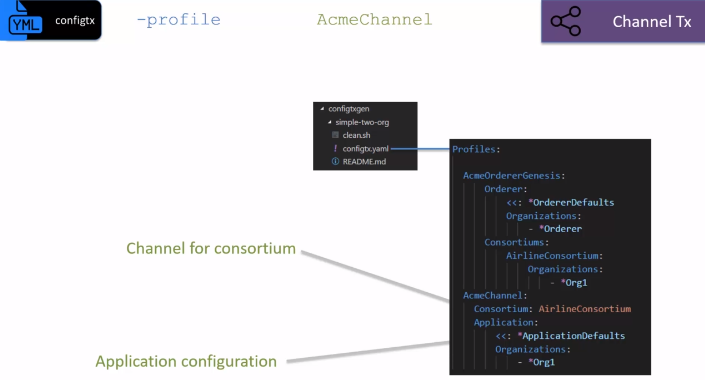


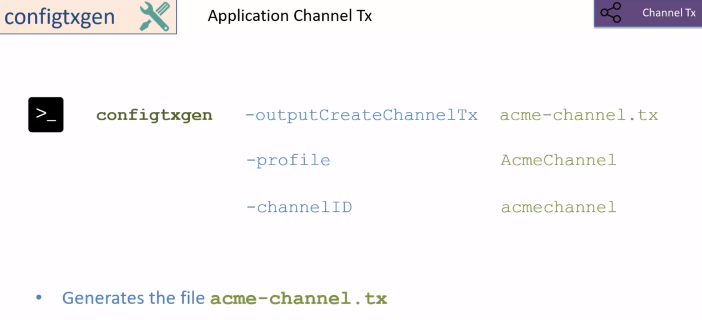
Now inspect this file



configtxgen -inspectBlock ./acme-genesis.block > temp/acme-genesis.json



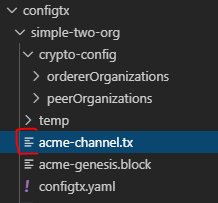


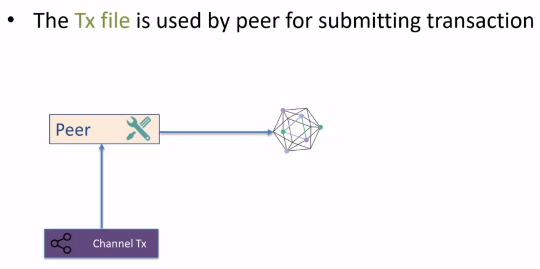


configtxgen -outputCreateChannelTx ./acme-channel.tx -profile AcmeChannel -channelID acmechannel

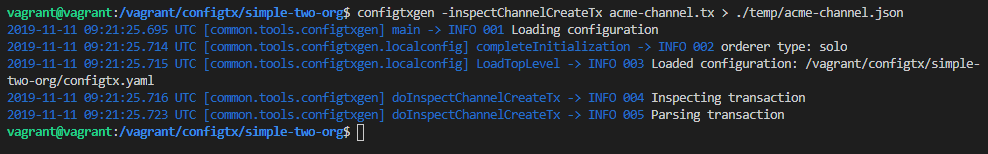


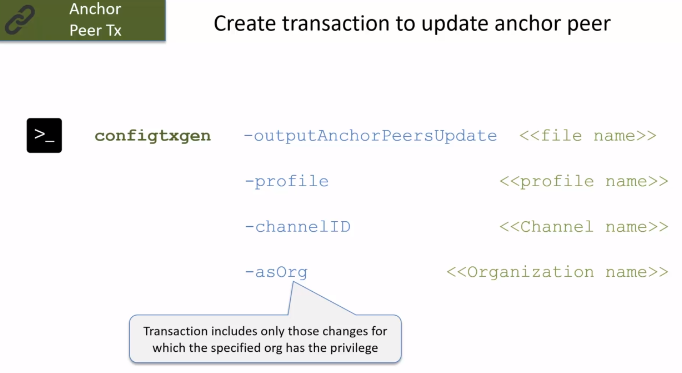
It will create the file





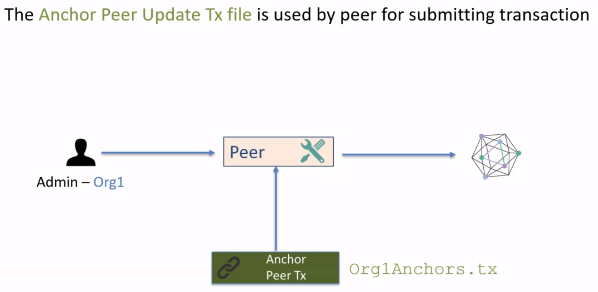
configtxgen -inspectChannelCreateTx acme-channel.tx > ./temp/acme-channel.json



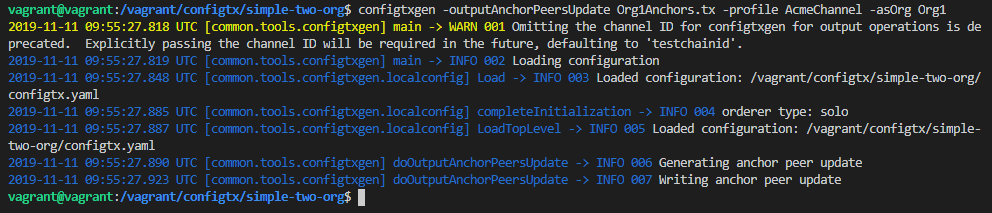




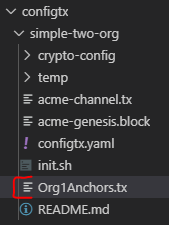




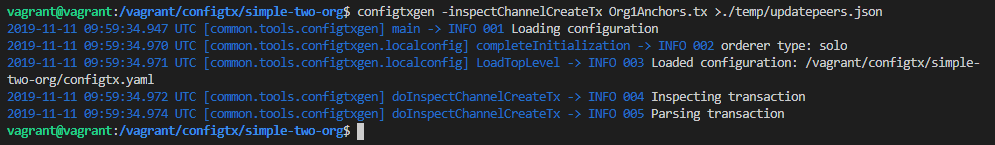
configtxgen -outputAnchorPeersUpdate Org1Anchors.tx -profile AcmeChannel -asOrg Org1

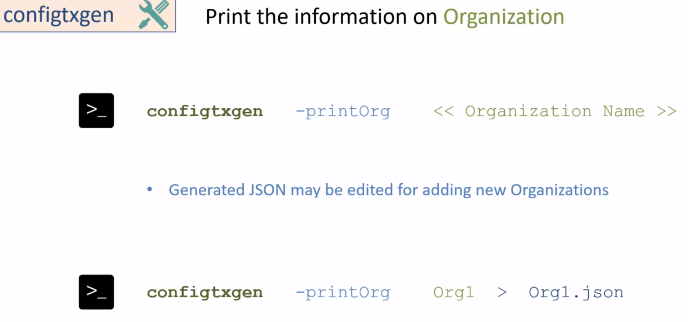


A new file will be created



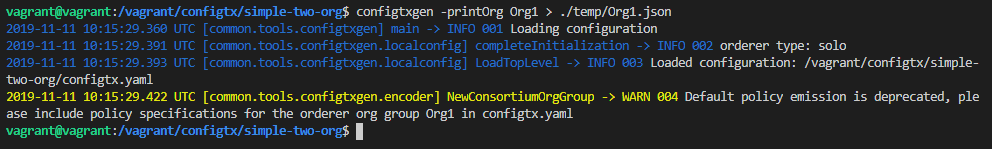
configtxgen -inspectChannelCreateTx Org1Anchors.tx >./temp/updatepeers.json



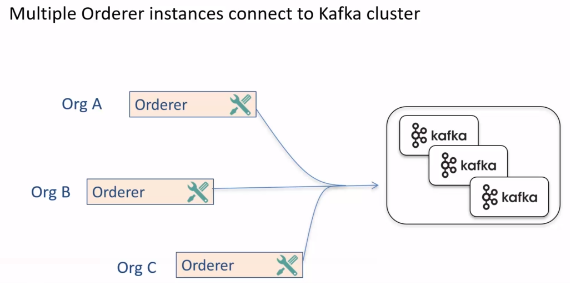


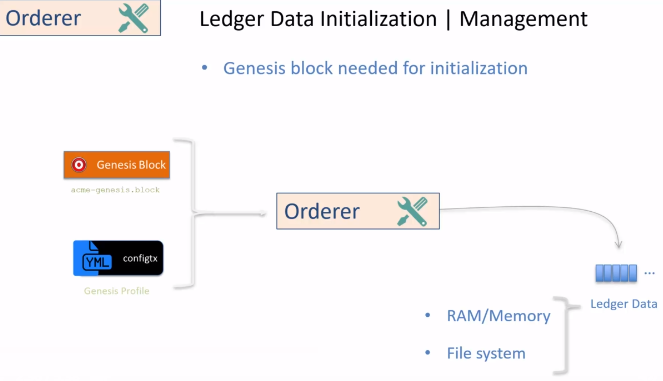
Print the organization information

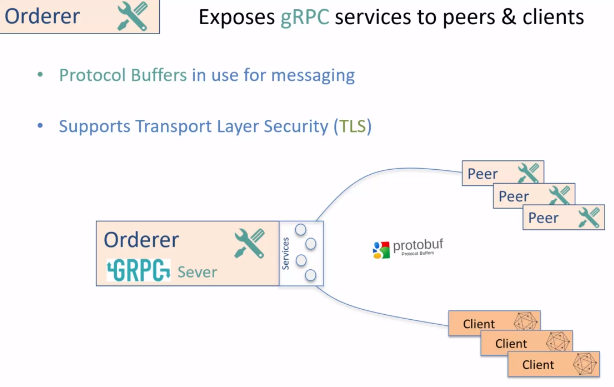
configtxgen -printOrg Org1 > ./temp/Org1.json



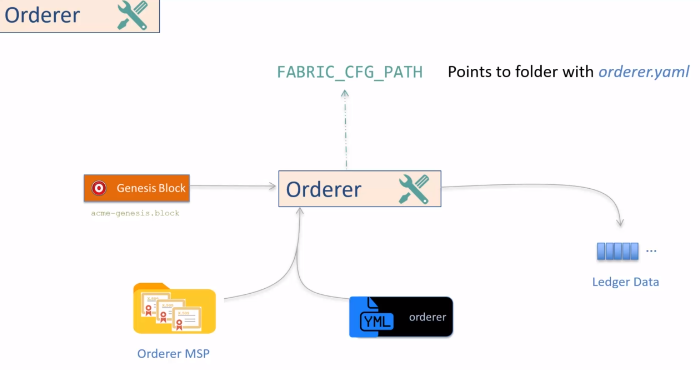
**Orderer Binary**

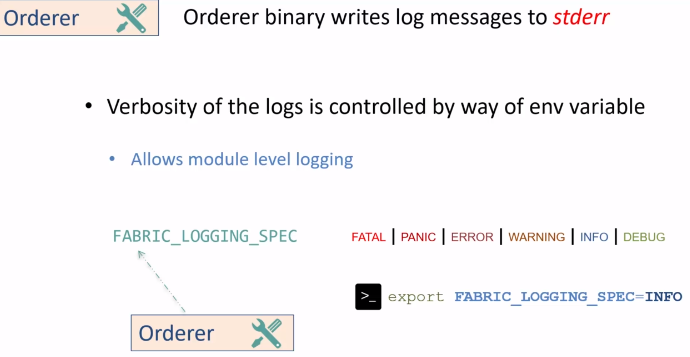


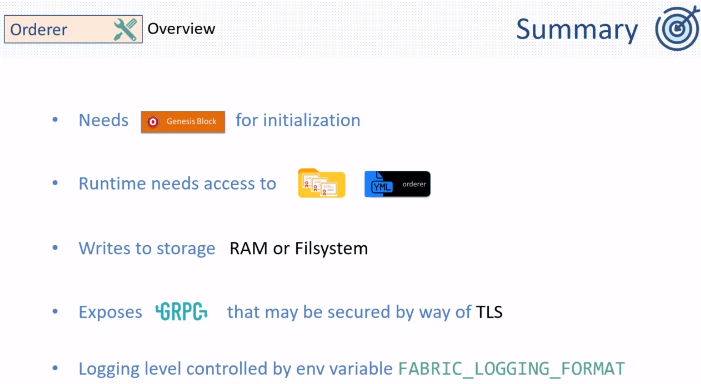


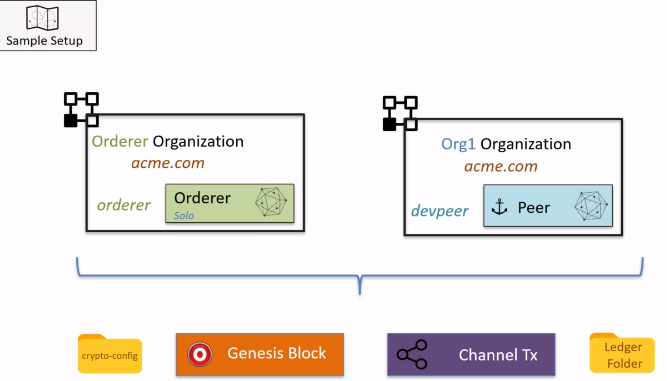


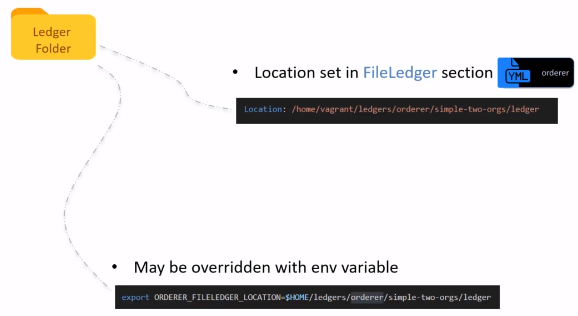


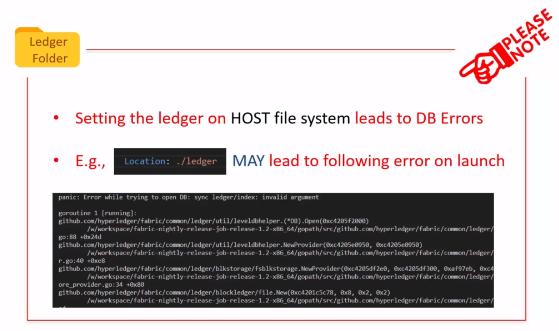


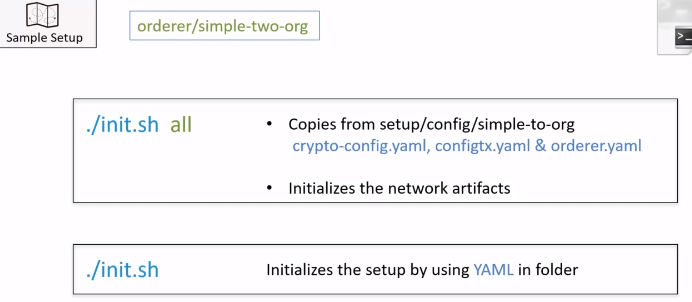


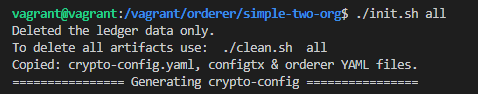




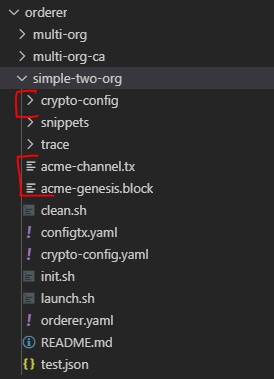








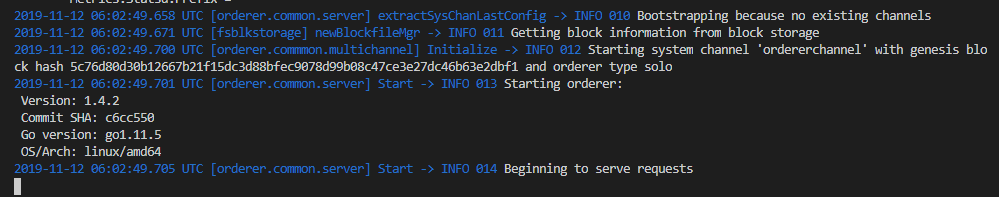
Following files & folders are generated





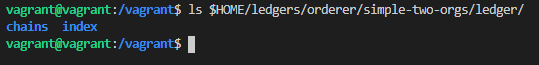


Orderer is started

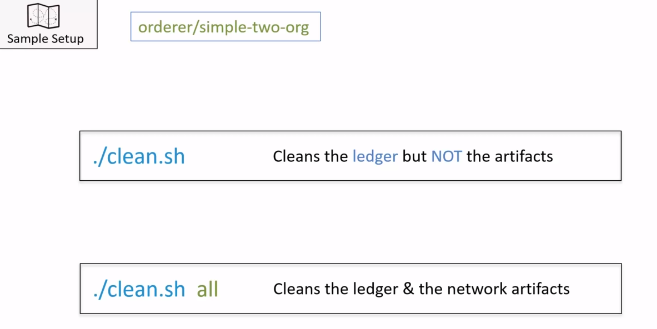


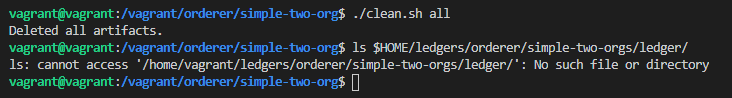
In 2nd terminal let’s see how the ledger is getting generated

ls $HOME/ledgers/orderer/simple-two-orgs/ledger/

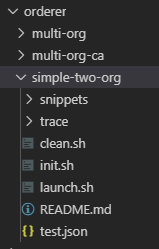


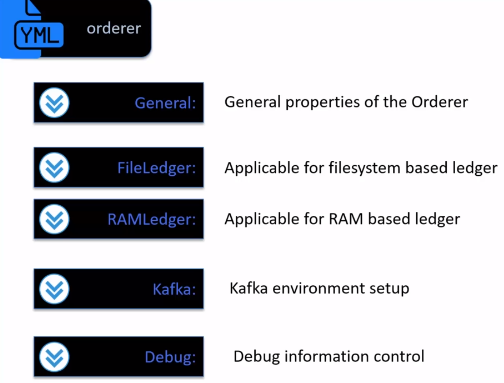
Now this is coming from the ledger that's been written by the orderer to the file system.

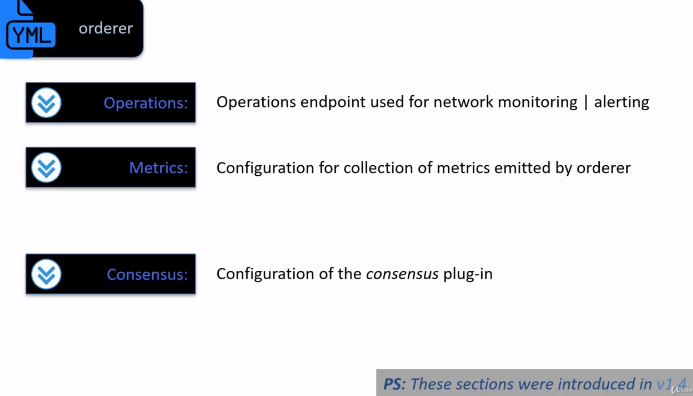


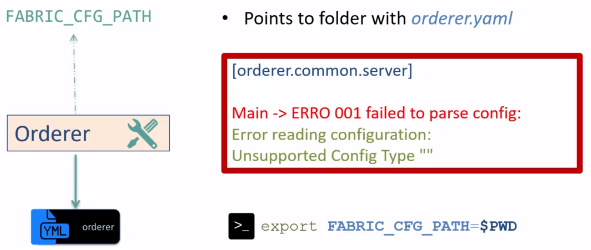


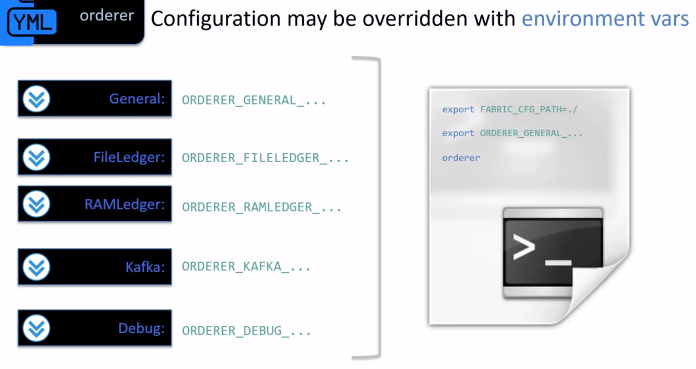
Deleted all artefacts and ledger folder cannot be found.

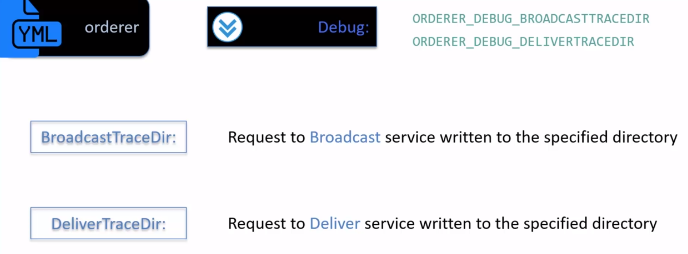


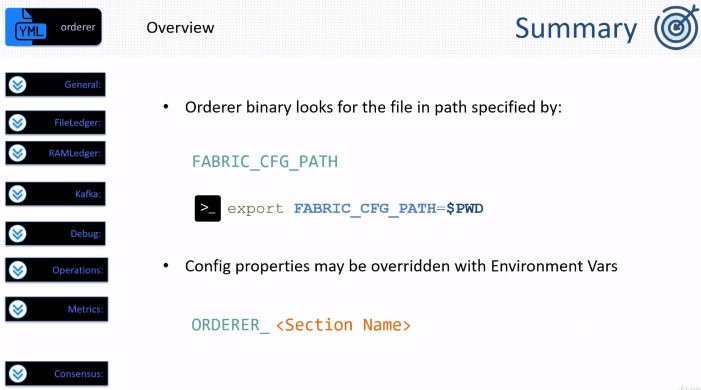


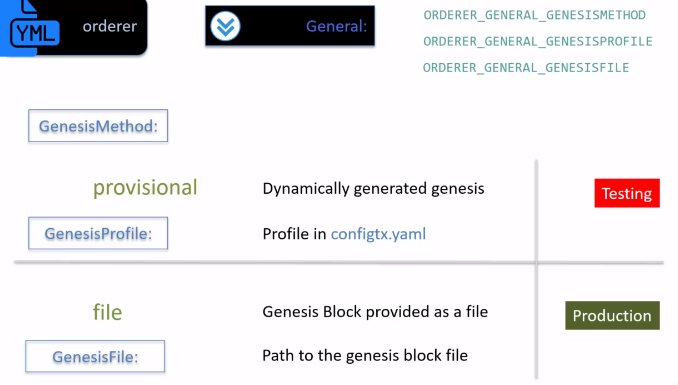




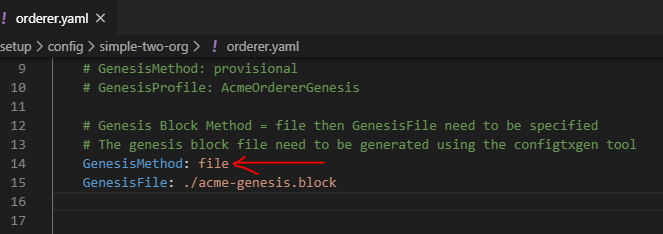








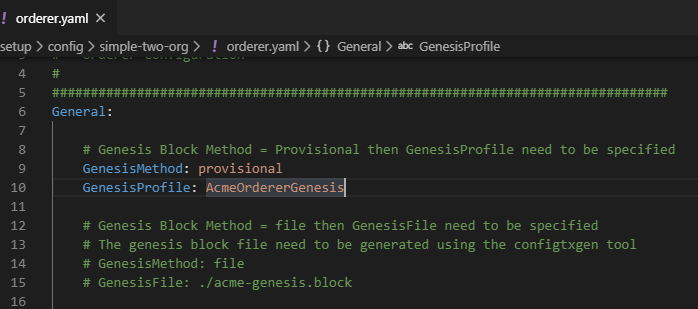
By default it is set to file



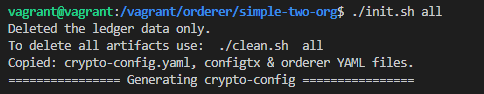
Acme-genesis.block file was created as part of the initialization.

Now use **GenesisMethod: provisional**

Comment and uncomment this

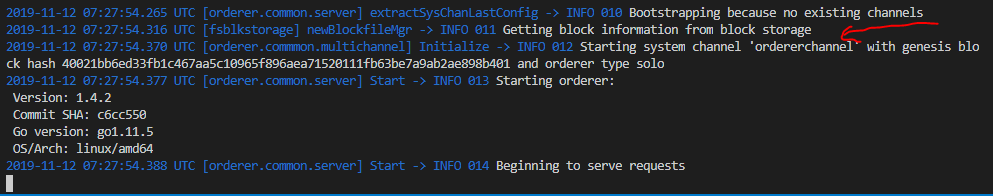


Now again run the init command like previous run





Take a look at the message with a provisional genesis block set up

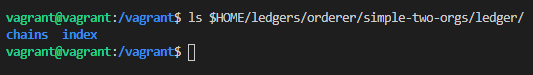


Starting system channel 'ordererchannel'.

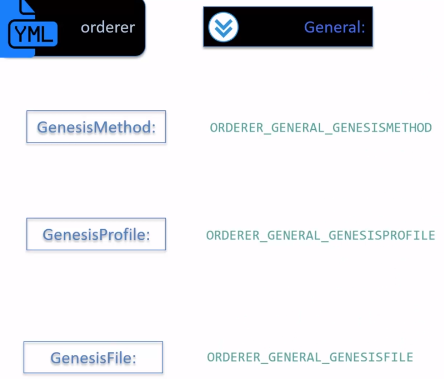
This is the name that orderer is giving to the temporary channel created with the provisional genesis block.

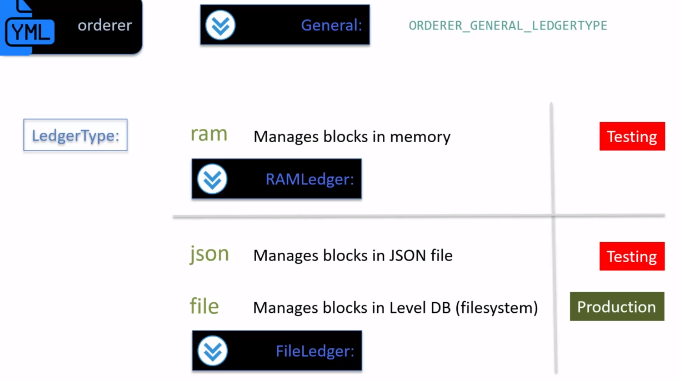
In 2nd terminal lets see the ledger folder

ls $HOME/ledgers/orderer/simple-two-orgs/ledger/

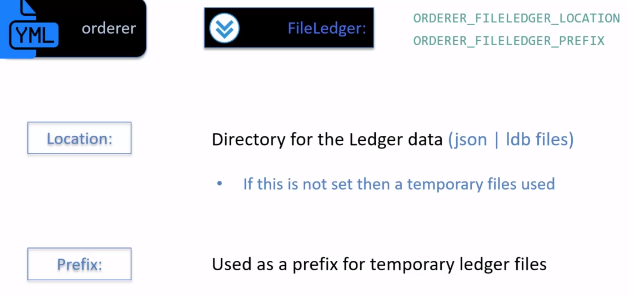


That means the orderer is up and writing for the ledger folder.

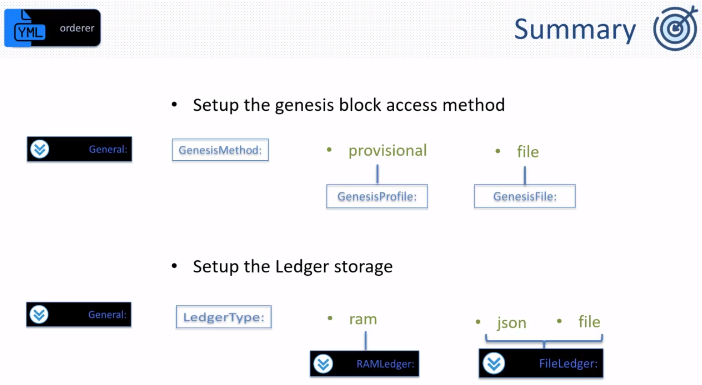


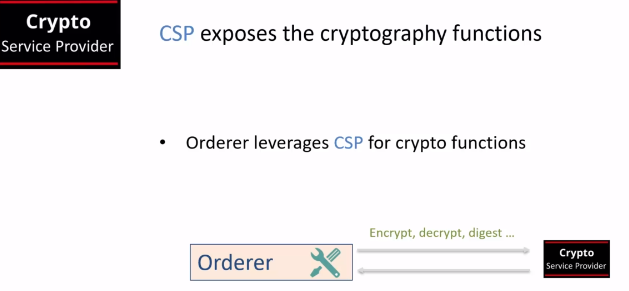


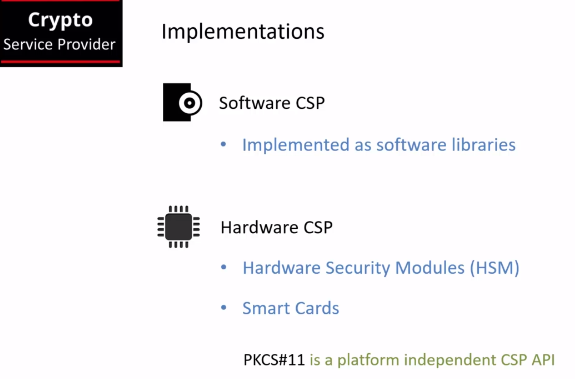


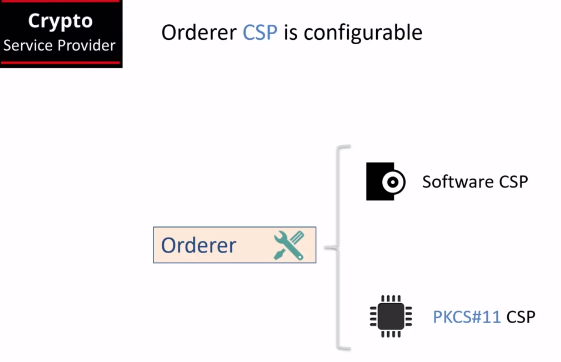


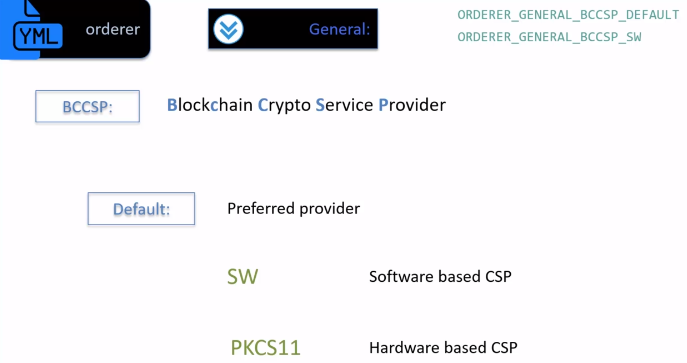
If Location is specified then prefix is ignored.

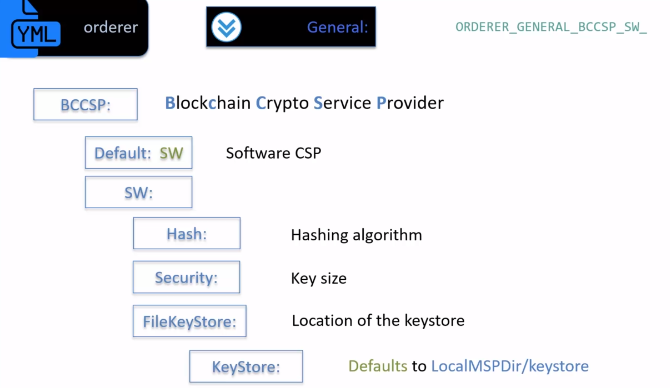




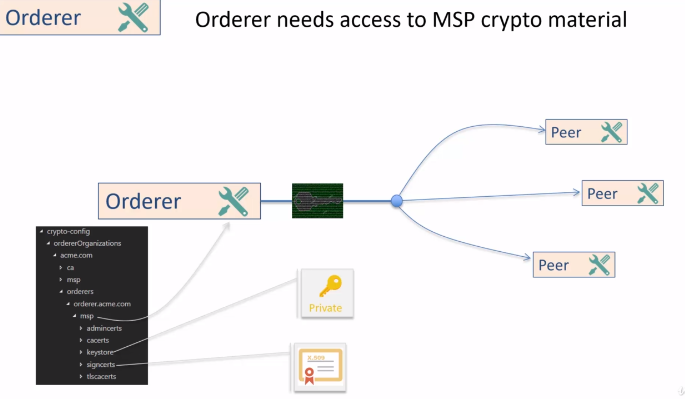


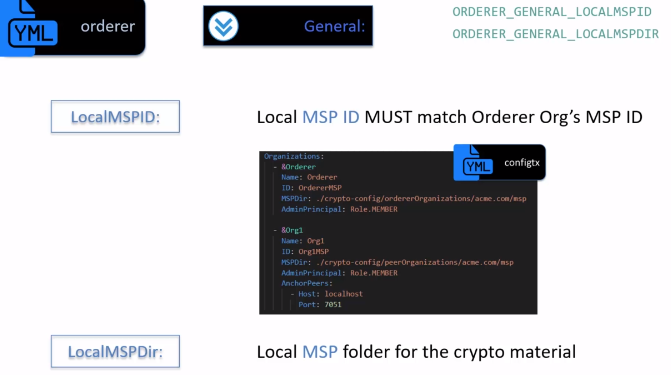


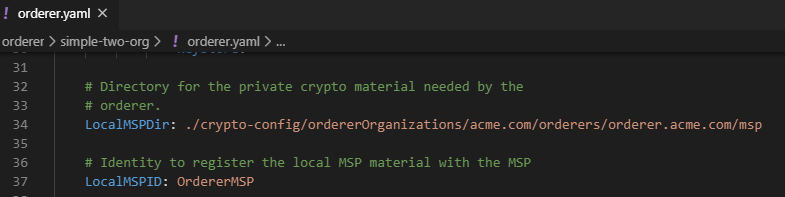


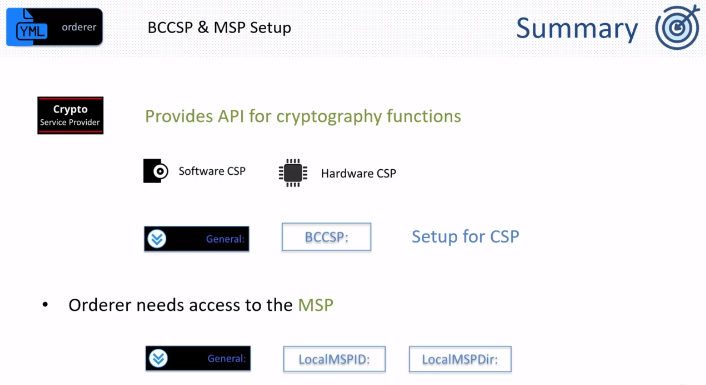


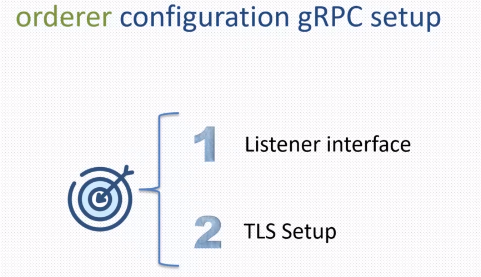


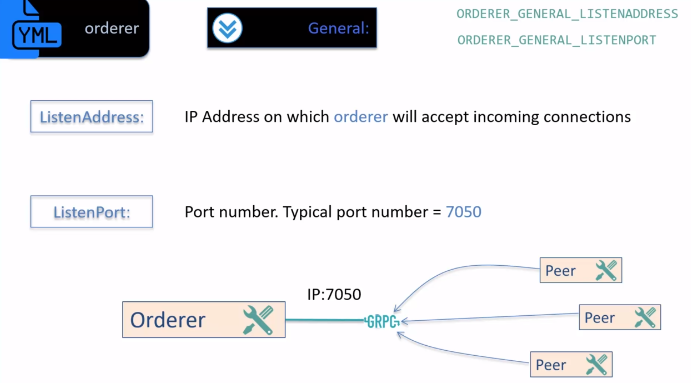


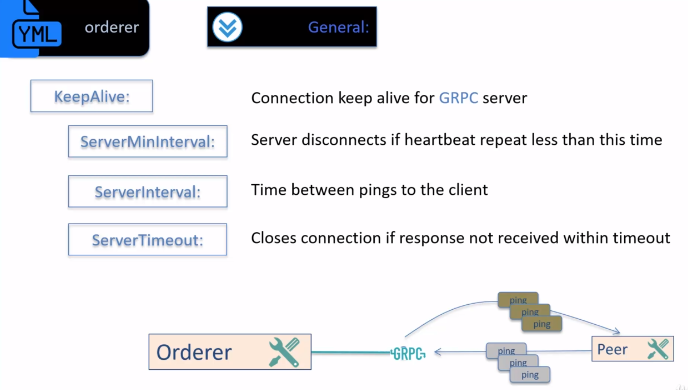


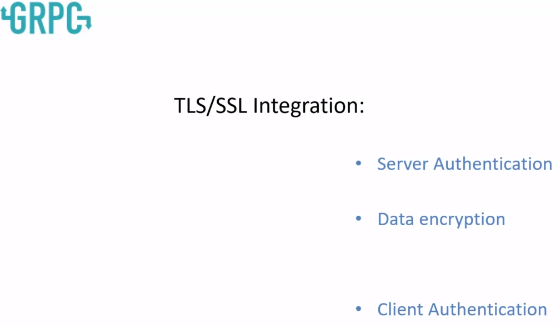


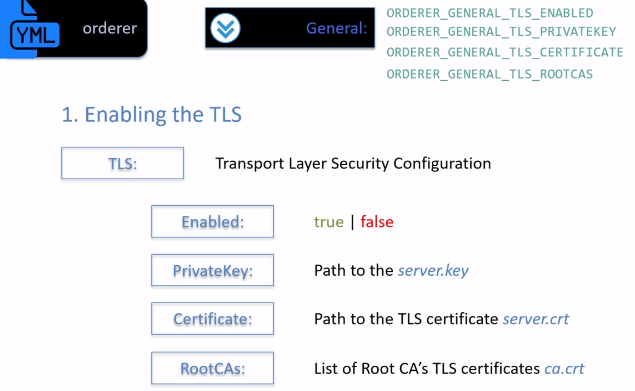


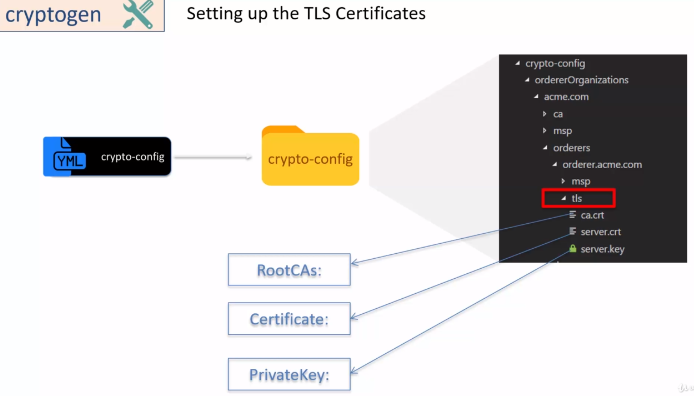


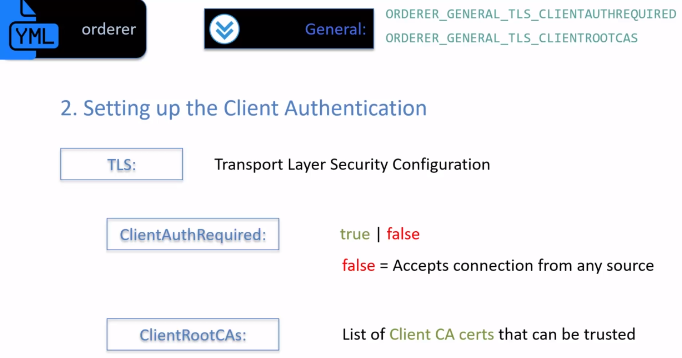


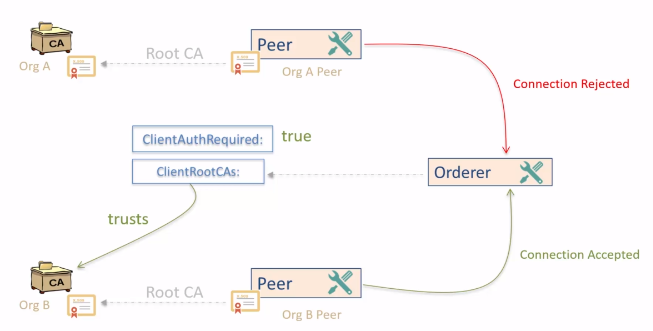






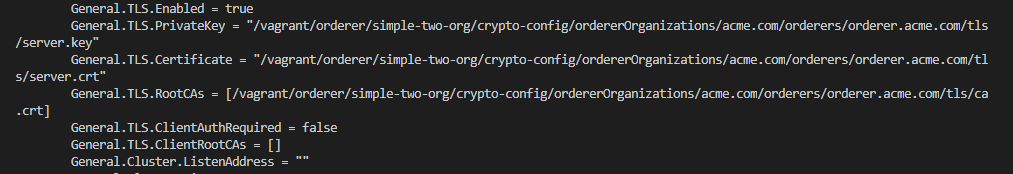




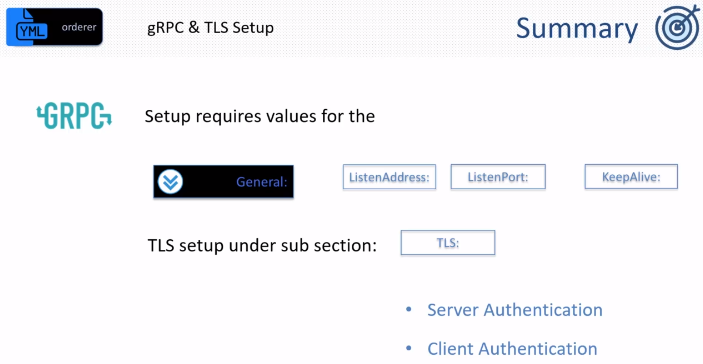


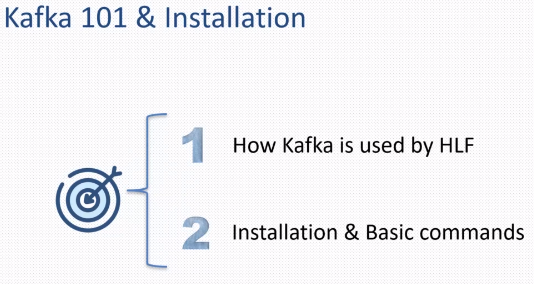


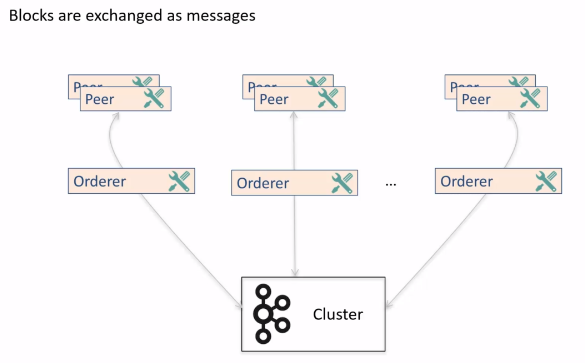
Once you run it you will find that the TLS enabled is set to true and all the values that we have provided in the environment variable for the private key certificate and rootCA is reflecting over here.

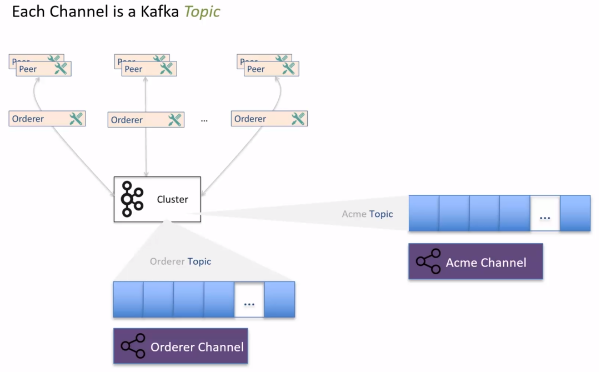


So at this point our orderer has launched with the TLS enabled.

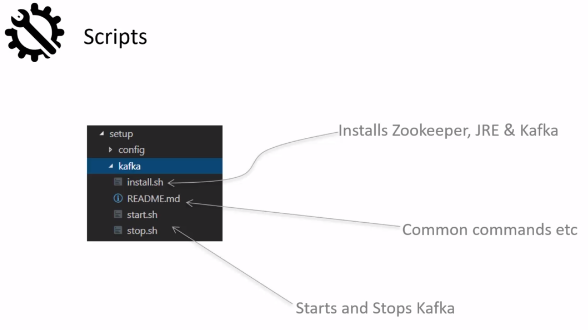


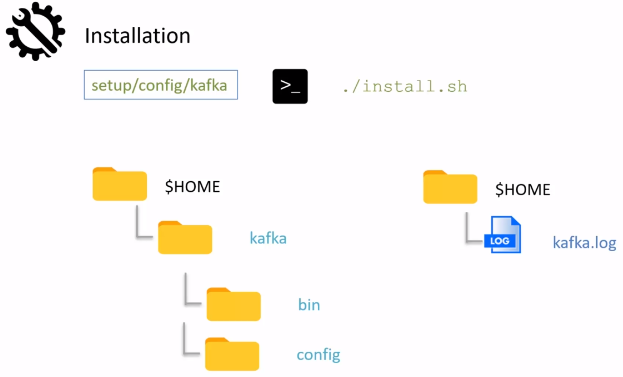




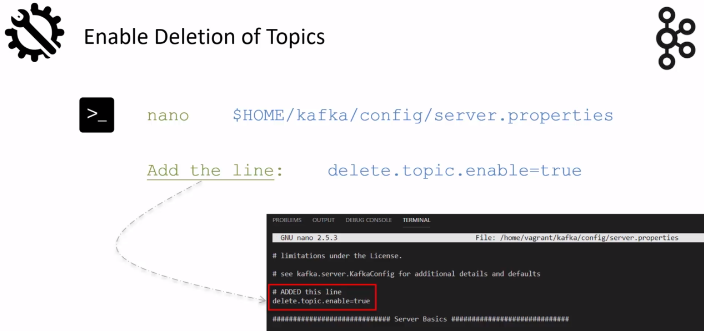


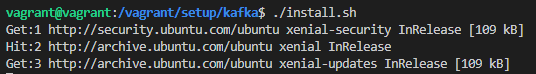


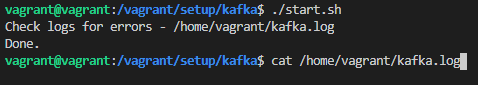


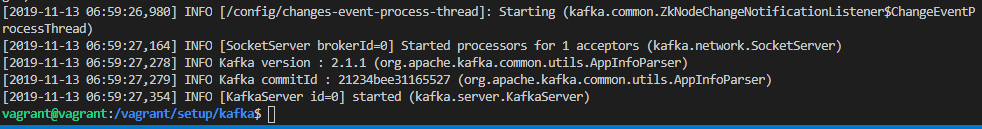


The default setup of Kafka does not allow deletion of topic. In order to enable deletion of topics, you need to make a manual change to the server.properties.

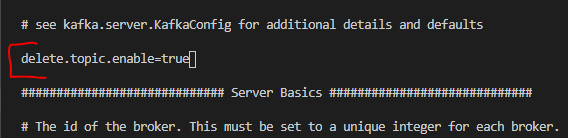


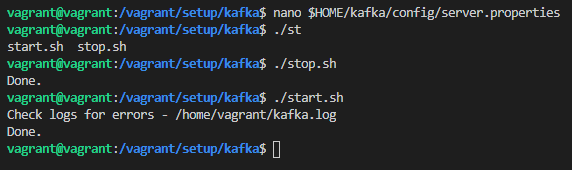












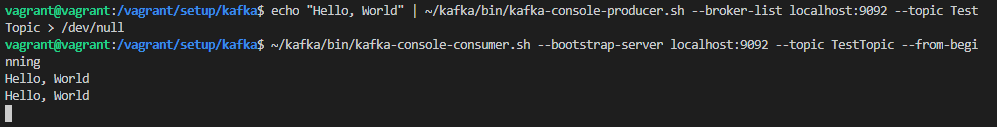
Now our server is enabled for the removal of topics as well.

Validate Kafka setup

I'm going to simply publish a message to the test topic and then execute the subscriber command on the test topic to see if the message sent with the publish, is received or not.

echo "Hello, World" | ~/kafka/bin/kafka-console-producer.sh --broker-list localhost:9092 --topic TestTopic > /dev/null

~/kafka/bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic TestTopic --from-beginning



And as you can see we have received the message Hello World.

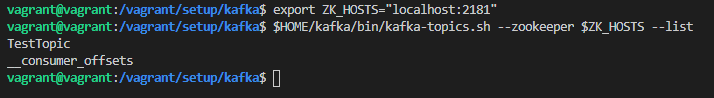
That was published as part of the publish command.

So our Kafka setup is now working.

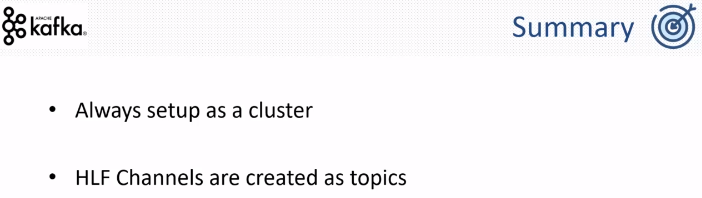
One such script is for the tropics management so to list all the topics for example you have to first set the zookeeper environment variable and then simply execute the script.

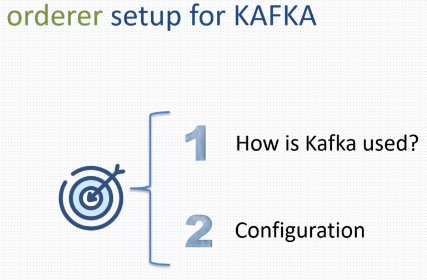
export ZK\_HOSTS="localhost:2181"

$HOME/kafka/bin/kafka-topics.sh --zookeeper $ZK\_HOSTS --list



Here as you can see the only topic is test topic which was created as part of the publish.





How orderer configured for the Blockchain network

