


Create a Cloud9 environment
IAM Configuration
Modify Cloud9 IAM role

- Set up a Fabric client
- ▼ Write and deploy chaincode
 - Chaincode development environment
 - Write chaincode
 - Create sharing policy
 - Create member identities
 - Configure main channel
 - Create main channel**
 - Join main channel
 - Build chaincode
 - Install chaincode
 - Approve and commit the chaincode
- Test the chaincode
- Congratulations

▼ AWS account access

[Open AWS console \(us-east-1\)](#) 

[Get AWS CLI credentials](#)

Exit event

Event ends in 25 minutes.

[Event dashboard](#) > [Write and deploy chaincode](#) > **Create main channel**

Create main channel

Only the **Retailer** should run the following commands from its Cloud9 terminal.

Bundle the member certificates together so that they can easily be downloaded and installed by each member. Remember, it is important that each member has access to the public keys of other members' client certificates in the channel. You created a shared S3 bucket in an earlier step to facilitate this exchange.

```
1 cd
2 mkdir -p $HOME/retailer-admin-msp/cacerts
3 mkdir -p $HOME/retailer-admin-msp/admincerts
4 mkdir -p $HOME/supplier-admin-msp/cacerts
5 mkdir -p $HOME/supplier-admin-msp/admincerts
6 aws s3api get-object --bucket $BUCKET_NAME --key rtcacert.pem $HOME/retailer-admin-msp/cacerts/cacert.pem
7 aws s3api get-object --bucket $BUCKET_NAME --key rtadmindcert.pem $HOME/retailer-admin-msp/admincerts/cert.pem
8 aws s3api get-object --bucket $BUCKET_NAME --key spcacert.pem $HOME/supplier-admin-msp/cacerts/cacert.pem
9 aws s3api get-object --bucket $BUCKET_NAME --key spadmindcert.pem $HOME/supplier-admin-msp/admincerts/cert.pem
10 tar czf certs.tgz retailer-admin-msp supplier-admin-msp
11 aws s3api put-object --bucket $BUCKET_NAME --key certs.tgz --body $HOME/certs.tgz --acl bucket-owner-full-control
```

Create a channel creation transaction for the supply chain network:

```
1 configtxgen -outputCreateChannelTx $HOME/mainchannel.pb -profile TwoOrgChannel -channelID mainchannel -configPath $HOME/
```

The output should look like:

```
1 2022-11-02 21:47:28.994 UTC [common.tools.configtxgen] main -> INFO 001 Loading configuration
2 2022-11-02 21:47:29.004 UTC [common.tools.configtxgen.localconfig] Load -> INFO 002 Loaded configuration: /home/ec2-user/configtx.yaml
3 2022-11-02 21:47:29.004 UTC [common.tools.configtxgen] doOutputChannelCreateTx -> INFO 003 Generating new channel configtx
4 2022-11-02 21:47:29.006 UTC [common.tools.configtxgen] doOutputChannelCreateTx -> INFO 004 Writing new channel tx
```

Check that the channel configuration has been generated:

```
1 ls -lt ~/mainchannel.pb
```

Create a Cloud9 environment
IAM Configuration
Modify Cloud9 IAM role

► Set up a Fabric client

▼ Write and deploy chaincode

Chaincode development environment

Write chaincode

Create sharing policy

Create member identities

Configure main channel

Create main channel

Join main channel

Build chaincode

Install chaincode

Approve and commit the chaincode

► Test the chaincode

Congratulations

▼ AWS account access

[Open AWS console](#)

(us-east-1) 

[Get AWS CLI credentials](#)

Exit event

The output of that `ls` command should look like:

```
1  -rw-r----- 1 ec2-user ec2-user 526 Nov  2 21:47 /home/ec2-user/mainchannel.pb
```



Then create the channel:

```
1  peer channel create -c mainchannel -f $HOME/mainchannel.pb -o $ORDERER --cafile $HOME/managedblockchain-tls-chain.pem --tls
```



The expected output should look like:

```
1  2020-10-16 02:54:43.177 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2  2020-10-16 02:54:47.537 UTC [cli.common] readBlock -> INFO 002 Received block: 0
```



Download the genesis block of the channel:

```
1  peer channel fetch oldest $HOME/mainchannel.block -c mainchannel -o $ORDERER --cafile $HOME/managedblockchain-tls-chain.pem --tls
```



The output should look like:

```
1  2020-07-23 03:59:13.501 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2  2020-07-23 03:59:13.506 UTC [cli/common] readBlock -> INFO 002 Received block: 0
```



Join the first peer to the channel:

```
1  peer channel join -b $HOME/mainchannel.block
```



The output should look like:

```
1  2020-07-23 03:59:21.777 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and orderer connections initialized
2  2020-07-23 03:59:21.978 UTC [channelCmd] executeJoin -> INFO 002 Successfully submitted proposal to join channel
```



Check whether the first peer has successfully joined the channel `mainchannel`

```
1  peer channel list
```



Create a Cloud9
environment

IAM Configuration

Modify Cloud9 IAM role

► Set up a Fabric client

▼ Write and deploy chaincode

Chaincode development
environment

Write chaincode

Create sharing policy

Create member identities

Configure main channel

Create main channel

Join main channel

Build chaincode

Install chaincode

Approve and commit the
chaincode

► Test the chaincode

Congratulations

▼ AWS account access

[Open AWS console
\(us-east-1\)](#) 

[Get AWS CLI credentials](#)

Exit event

You should receive the output below:

```
1 Channels peers has joined:
2 mainchannel
```



Join the second peer to the channel:

```
1 CORE_PEER_ADDRESS=$PEER2ENDPOINT peer channel join -b $HOME/mainchannel.block
```



Check whether the second peer has successfully joined the channel `mainchannel`

```
1 peer channel list
```



You should receive the output below:

```
1 Channels peers has joined:
2 mainchannel
```



Previous

Next

