< 하이퍼레져 페브릭5 >

1. Private Data

- 비트코인, 이더리움에서 사용했던 consensus(합의)라는 단어 대신 policy(정책)이란 단어 사용!
- blockToLive라는 개념을 도입해 일정블록 생성 이후 데이터 삭제 가능하게 만듬(보호를 위해)
- collectionMarbles, collectionMarblesPrivateDetails라는 JSON파일

- fabric-samples/chaincode/marbles02_private 밑에 있는 파일 두 개 열기(.json, .go)
  $ rmate -p 52698 collections_config.json
  $ rmate -p 52698 marbles_chaincode_private.go

collections_config.json | marbles_chaincode_private.go

```
[
  {
      "name": "collectionMarbles",
      "policy": "OR('Org1MSP.member', 'Org2MSP.member')",
      "requiredPeerCount": 0,
      "maxPeerCount": 3,
      "blockToLive":1000000
  },
  {
      "name": "collectionMarblePrivateDetails",
      "policy": "OR('Org1MSP.member')",
      "requiredPeerCount": 0,
      "maxPeerCount": 3,
      "blockToLive":3
  }
]
```

- **$ ./byfn.sh down**으로 모든 채널,서버 죽이고
  $ ./byfn.sh up –c mychannel –s couchdb를 이용해 couchdb를 갖는 새로운 시스템 기동
- **$ docker exec –it cli bash**를 이용해 docker로 이동!!
  interprete terminal
- 각 peer별 체인코드(marblesp) install 후 instantiate(이놈은 한 번만!)
  기존과 똑같이 치고 마지막에 —collections-config 옵션 추가(젤 중요★)
- 자주 나오는 포트번호의 의미? 7050(orderer), 그 이상(각종 peer)
- **$ peer chaincode invoke -o orderer.example.com:7050 —tls –cafile /opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem –C mychannel -n marblesp –c '{"Args":["initMarble","marble1","blue","35","tom","99"]}'**

```
func (t *SimpleChaincode) initMarble(stub shim.ChaincodeStubInterface, args []string) pb.Response {
    var err error

    //  0-name  1-color  2-size  3-owner  4-price
    // "asdf",  "blue",  "35",   "bob",   "99"
    if len(args) != 5 {
        return shim.Error("Incorrect number of arguments. Expecting 5")
    }
```

: initMarble이란 함수는 5개의 인자를 받음
(맨 위에 설정한 struct를 보면 마지막 price는 marblePrivateDetails의 데이터임을 알 수 있음

```
type marble struct {
    ObjectType string `json:"docType"`
    Name       string `json:"name"`
    Color      string `json:"color"`
    Size       int    `json:"size"`
    Owner      string `json:"owner"`
}

type marblePrivateDetails struct {
    ObjectType string `json:"docType"`
    Name       string `json:"name"`
    Price      int    `json:"price"`
}
```

코드 내에 PutState를 보면 4개 / 1개 값 따로 입력!!

: 어쨌든 위에 코드 실행(invoke)하면 아래 결과 표시

```
2018-11-20 07:09:32.224 UTC [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200
```

- initMarble가 잘 실행되었는지 Query문을 통해 확인(Org1에서 2개 → Org2에서 2개)

: **$ peer chaincode query -C mychannel -n marblesp -c '{"Args":["readMarble","marble1"]}'**

```
root@10cc4270dc10:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode query -C mychannel -n marblesp -c '{"Args":["readMarble","marble1"]}'
{"color":"blue","docType":"marble","name":"marble1","owner":"tom","size":35}
```

: **$ peer chaincode query -C mychannel -n marblesp -c '{"Args":["readMarblePrivateDetails","marble1"]}'**

```
root@10cc4270dc10:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode query -C mychannel -n marblesp -c '{"Args":["readMarblePrivateDetails","marble1"]}'
{"docType":"marblePrivateDetails","name":"marble1","price":99}
```

: Org1에서는 둘 다 잘 나오지만, Org2는 PrivateDetails(즉, price 정보)를 볼 수 없기 때문에 에러 발생!

```
root@10cc4270dc10:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode query -C mychannel -n marblesp -c '{"Args":["readMarblePrivateDetails","marble1"]}'
Error: endorsement failure during query. response: status:500 message:"{\"Error\":\"Failed to get private details for marble1: GET_STATE failed: transaction ID: dbcb417c2e11014fddff9c338de71fdab9fbac7a9e01bff8e481ae529150527c: private data matching public hash version is not available. Public hash version = &version.Height{BlockNum:0x6, TxNum:0x0}, Private data version = (*version.Height)(nil)\"}"
```

- 수명을 가지는 Private Data

: 새 터미널 창을 띄우고 **$ docker logs peer0.org1.example.com 2>&1 | grep -i -a –E 'private|pvt|privdata'** 입력해 현재 몇 개의 블록이 만들어졌는지 확인

```
guru@guru:~$ docker logs peer0.org1.example.com 2>&1 | grep -i -a -E 'private|pvt|privdata'
2018-11-20 06:02:00.749 UTC [kvledger] CommitWithPvtData -> INFO 02e [mychannel] Committed block [0] with 1 transaction(s) in 110ms (state_validation=0ms block_commit=15ms state_commit=92ms)
2018-11-20 06:02:13.871 UTC [gossip/privdata] StoreBlock -> INFO 03b [mychannel] Received block [1] from buffer
2018-11-20 06:02:13.935 UTC [kvledger] CommitWithPvtData -> INFO 042 [mychannel] Committed block [1] with 1 transaction(s) in 38ms (state_validation=0ms block_commit=16ms state_commit=19ms)
2018-11-20 06:02:17.058 UTC [gossip/privdata] StoreBlock -> INFO 043 [mychannel] Received block [2] from buffer
2018-11-20 06:02:17.222 UTC [kvledger] CommitWithPvtData -> INFO 04a [mychannel] Committed block [2] with 1 transaction(s) in 124ms (state_validation=0ms block_commit=58ms state_commit=63ms)
2018-11-20 06:02:57.835 UTC [gossip/privdata] StoreBlock -> INFO 04f [mychannel] Received block [3] from buffer
2018-11-20 06:02:58.178 UTC [kvledger] CommitWithPvtData -> INFO 053 [mychannel] Committed block [3] with 1 transaction(s) in 286ms (state_validation=17ms block_commit=62ms state_commit=202ms)
2018-11-20 06:03:39.669 UTC [gossip/privdata] StoreBlock -> INFO 05a [mychannel] Received block [4] from buffer
2018-11-20 06:03:39.818 UTC [kvledger] CommitWithPvtData -> INFO 05c [mychannel] Committed block [4] with 1 transaction(s) in 139ms (state_validation=50ms block_commit=39ms state_commit=46ms)
2018-11-20 06:42:02.137 UTC [gossip/privdata] StoreBlock -> INFO 060 [mychannel] Received block [5] from buffer
2018-11-20 06:42:02.829 UTC [kvledger] CommitWithPvtData -> INFO 065 [mychannel] Committed block [5] with 1 transaction(s) in 655ms (state_validation=499ms block_commit=53ms state_commit=93ms)
2018-11-20 07:09:34.254 UTC [gossip/privdata] StoreBlock -> INFO 06d [mychannel] Received block [6] from buffer
2018-11-20 07:09:34.496 UTC [couchdb] CreateDatabaseIfNotExist -> INFO 06f Created state database mychannel_marblesp$$pcollection$marble$private$details
2018-11-20 07:09:34.572 UTC [couchdb] CreateDatabaseIfNotExist -> INFO 070 Created state database mychannel_marblesp$$hcollection$marble$private$details
2018-11-20 07:09:34.877 UTC [kvledger] CommitWithPvtData -> INFO 071 [mychannel] Committed block [6] with 1 transaction(s) in 560ms (state_validation=42ms block_commit=70ms state_commit=440ms)
```

: 현재까지 6개의 블록이 생성됨! 'collections_config.json' 파일에 명시한 내용에 의해 앞으로 3개의 블록을 더 생성하면 price라는 data는 사라질 것!(실습을 위해 invoke – initMarble 3번 실시)

: initMarble(7번 블록) → transferMarble(8번 블록) → transferMarble(9번 블록) 할 때 까지 query문 실행해보면 이상 없이 'price:99'라는 결과가 호출!!

: transferMarble(10번 블록) 마지막 실행 후 query 날려보면 데이터 삭제된 것을 알 수 있음.

```
root@10cc4270dc10:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode query -C mychannel -n marblesp -c '{"Args":["readMarblePrivateDetails","marble1"]}'
Error: endorsement failure during query. response: status:500 message:"{\"Error\":\"Marble private details does not exist: marble1\"}"
```

```
t=18ms state_commit=237ms)
2018-11-20 08:18:15.825 UTC [gossip/privdata] StoreBlock -> INFO 08f [mychannel] Received block [10] from buffer
2018-11-20 08:18:15.378 UTC [kvledger] CommitWithPvtData -> INFO 091 [mychannel] Committed block [10] with 1 transaction(s) in 513ms (state_validation=38ms block_comm
```

: channel state가 아닌 private state에 저장되어 있다가 사라짐(json파일에 따로 policy를 설정해주면 사용 가능!)
sideDB(peer가 가지고 있는 개별적인 공간)