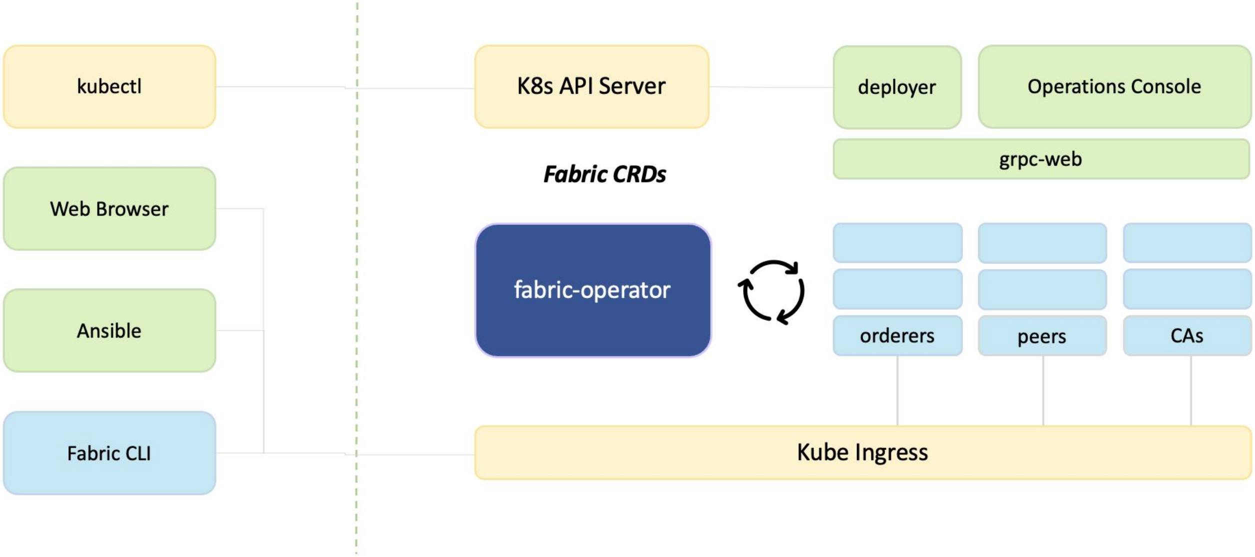
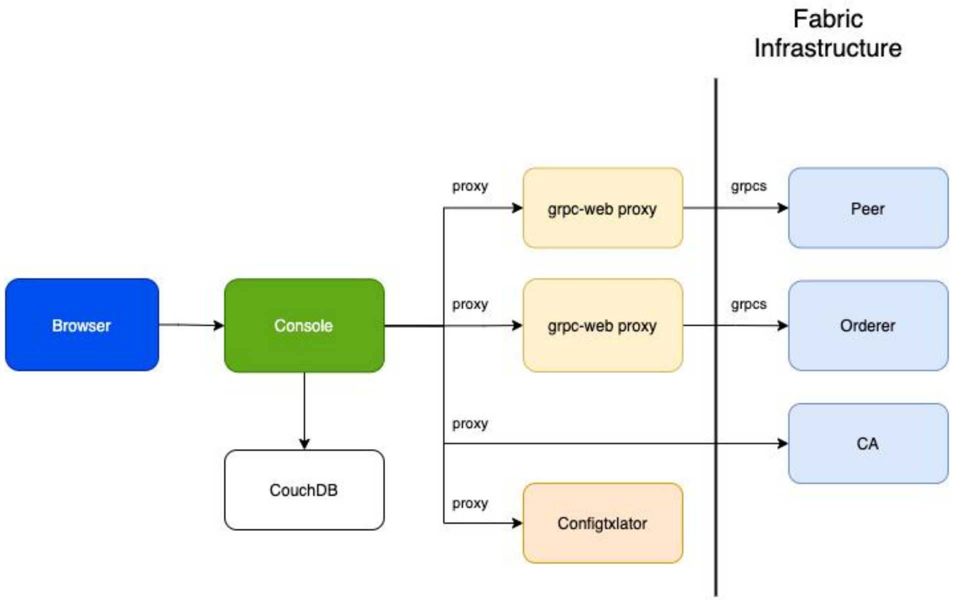
当前架构

不足

- 独立的账户体系不适合云原生环境
  - 缺少适配云原生环境的账户、角色、权限配置体系
- CRDs程度不足，无法充分利用云原生技术优势
  - 现有CRD没有充分使用
  - 缺少更多功能性CRDs
- 缺少监控运维体系、自动化部署

......





“

01

U4A结合

”

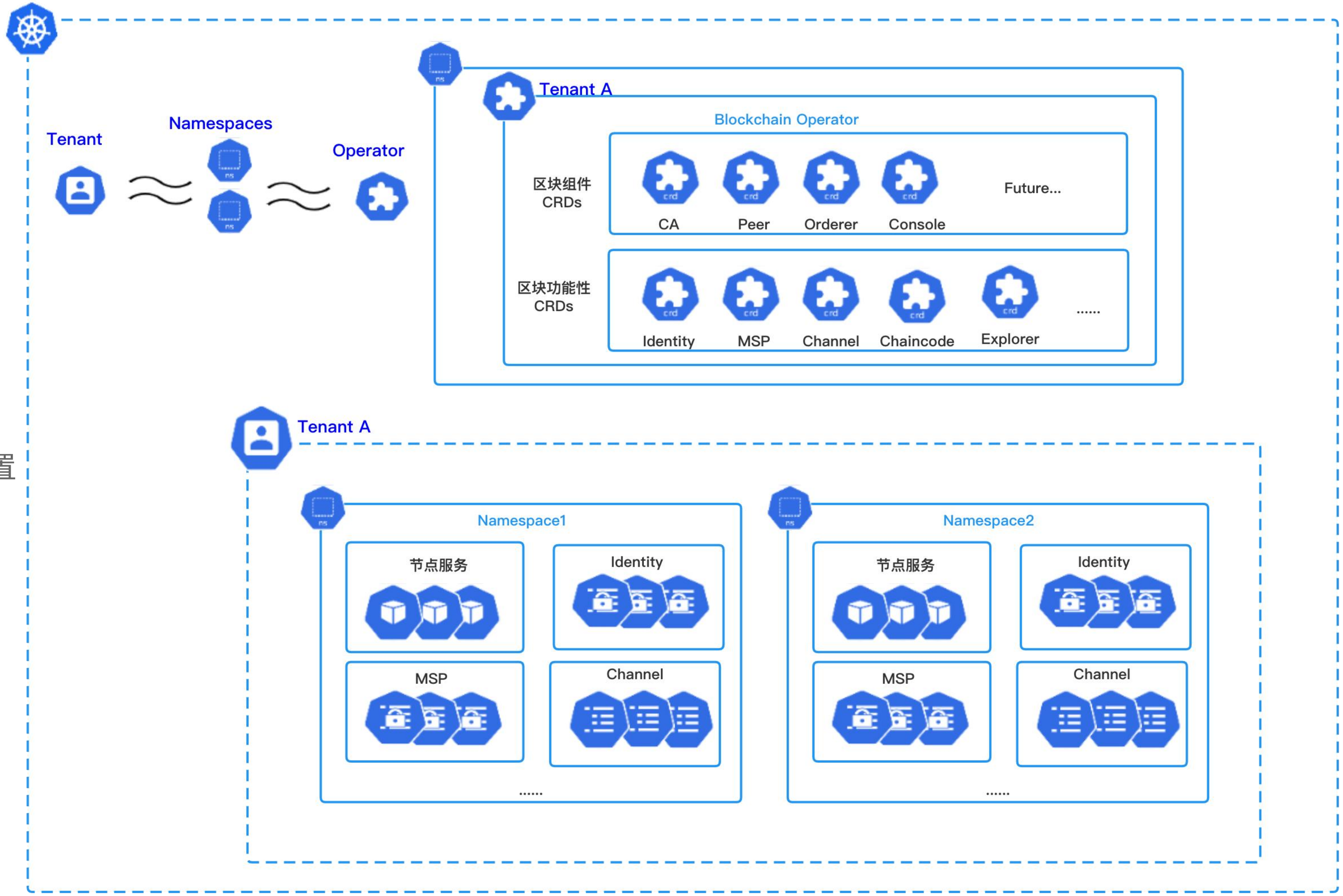
实施方案

租户定义：

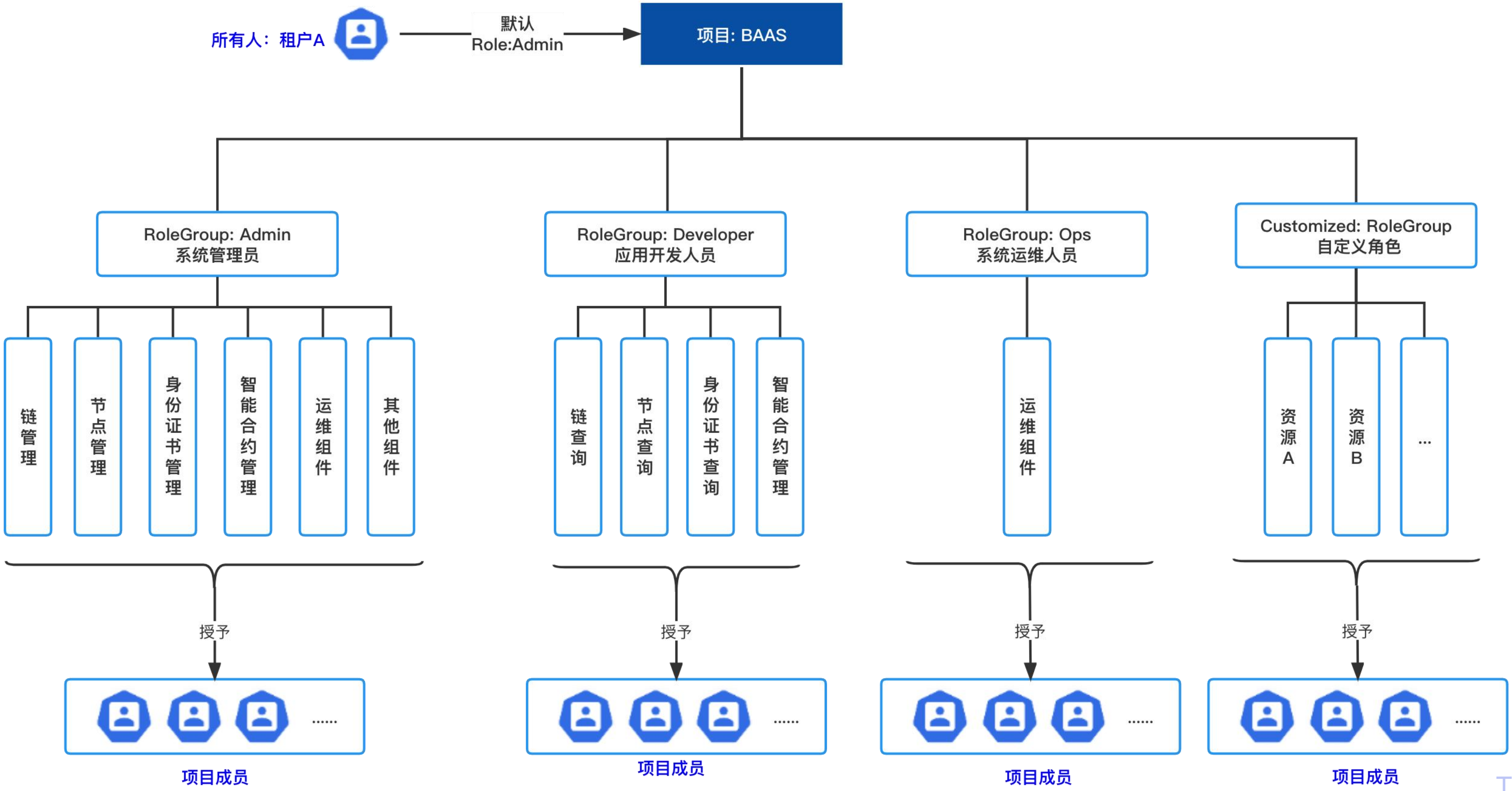
- 独立的个人、企业、机构
- 开通TCS服务
- 具备多namepsaces权限

实施：

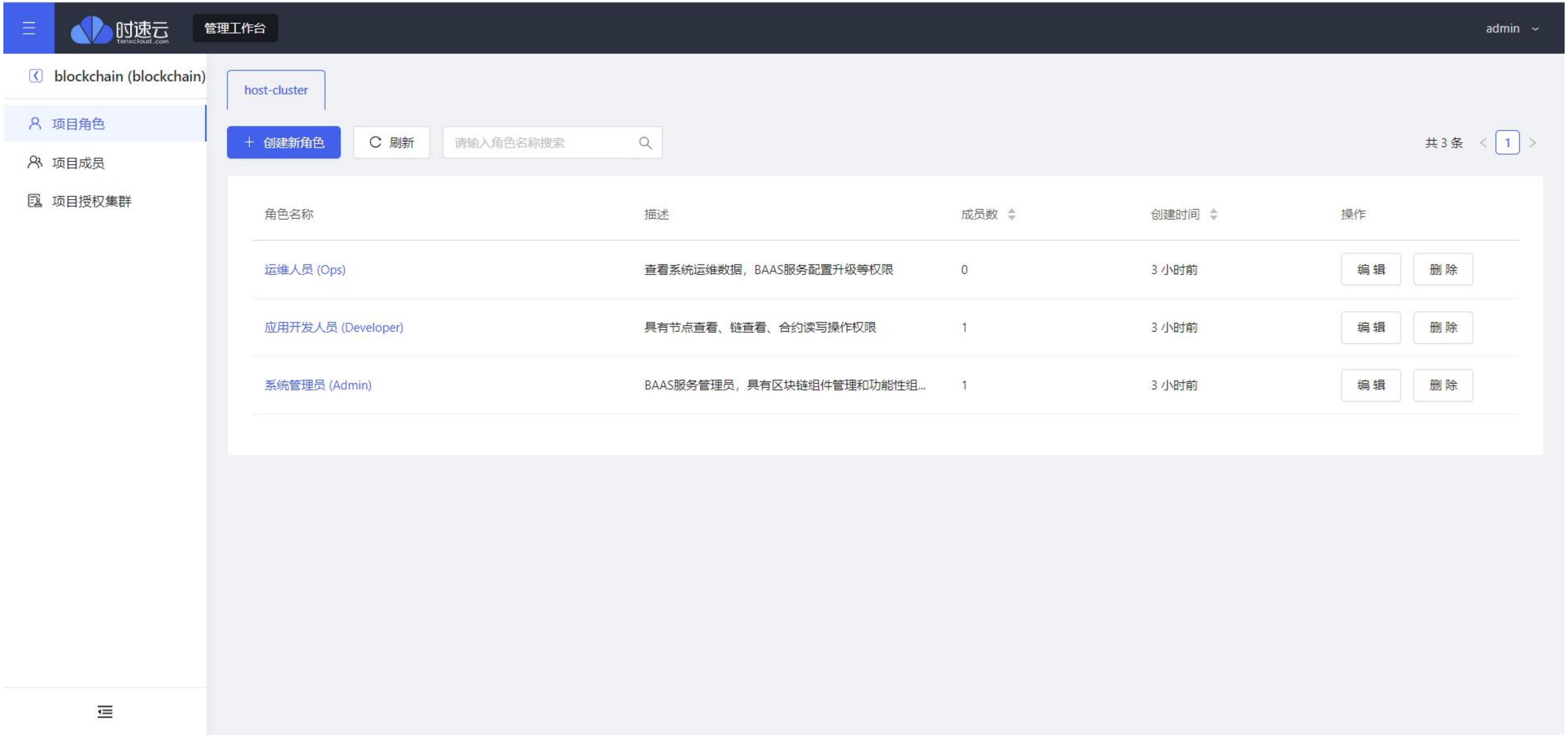
- 租户与BaaS组件 1： 1 独立部署
- Operator作用于租户授权namespaces
- 租户默认授予BAAS_ADMIN角色权限
- U4A提供统一的租户、成员组、权限配置服务（RBAC）



权限设计（U4A）

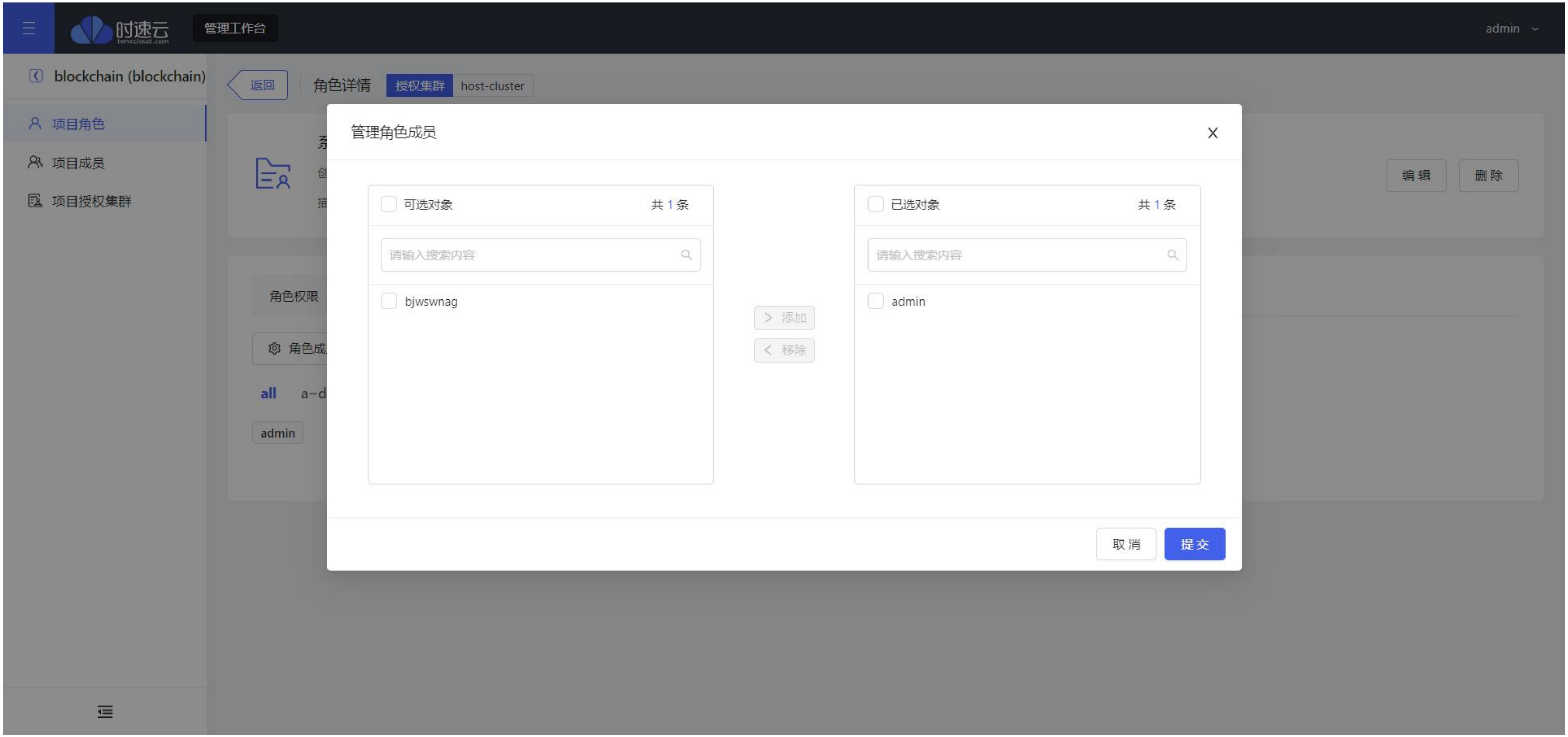

所有人：租户A
默认
Role:Admin
项目: BAAS
RoleGroup: Admin
系统管理员
RoleGroup: Developer
应用开发人员
RoleGroup: Ops
系统运维人员
Customized: RoleGroup
自定义角色
链管理
节点管理
身份证书管理
智能合约管理
运维组件
其他组件
链查询
节点查询
身份证书查询
智能合约管理
运维组件
资源A
资源B
...
授予
授予
授予
授予
......
......
......
......
项目成员
项目成员
项目成员
项目成员

权限设计（U4A）



角色名称	描述	成员数	创建时间	操作
运维人员 (Ops)	查看系统运维数据，BAAS服务配置升级等权限	0	3 小时前	编 辑 删 除
应用开发人员 (Developer)	具有节点查看、链查看、合约读写操作权限	1	3 小时前	编 辑 删 除
系统管理员 (Admin)	BAAS服务管理员，具有区块链组件管理和功能性组...	1	3 小时前	编 辑 删 除

权限设计（U4A）



管理角色成员

□ 可选对象 共 1 条

请输入搜索内容

□ bjwswnag

> 添加

< 移除

□ 已选对象 共 1 条

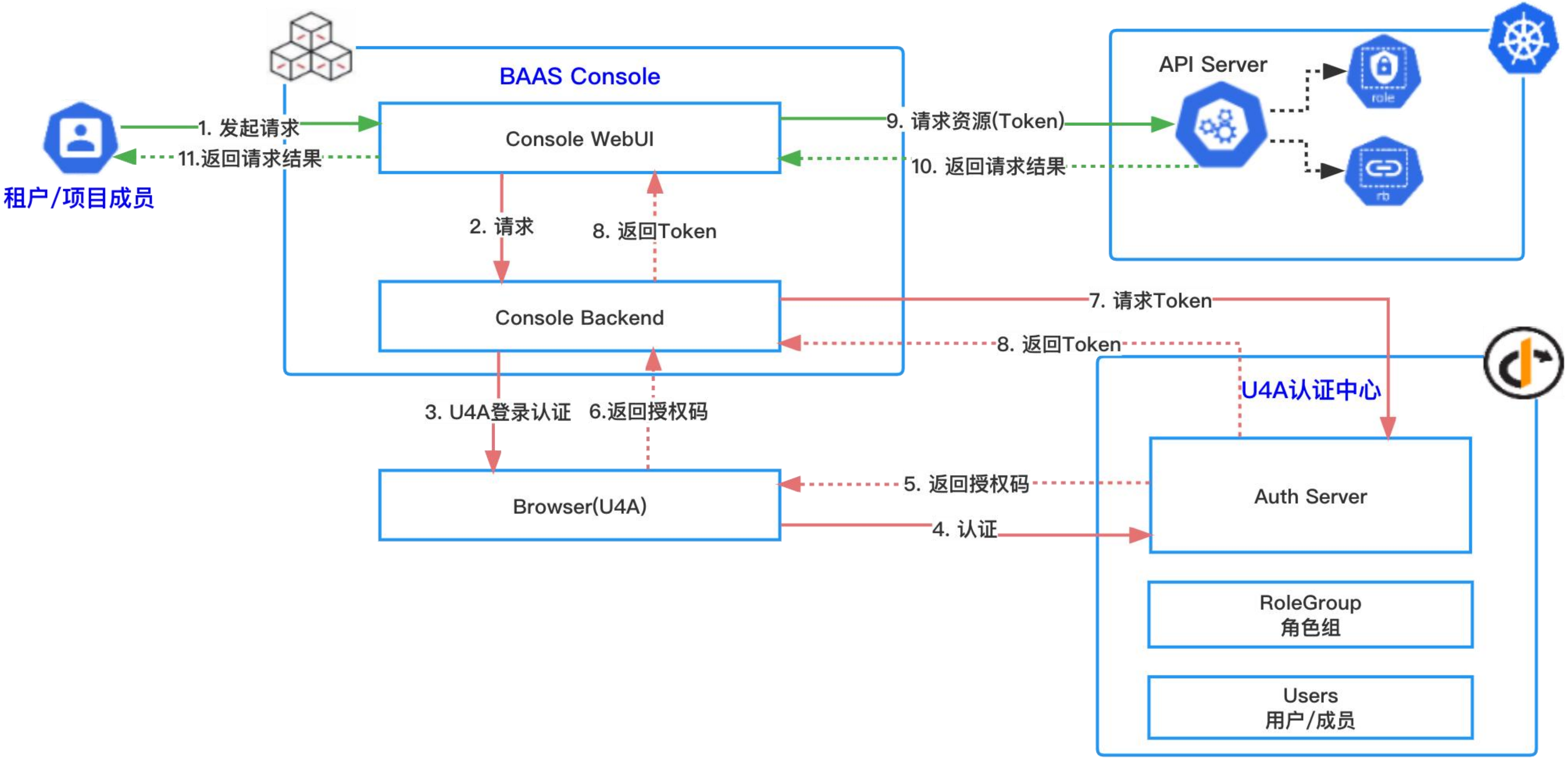
请输入搜索内容

□ admin

取 消 提 交

TenxCloud
时速云

核心流程（U4A）



TenxCloud
时速云

“02 扩充CRDs”

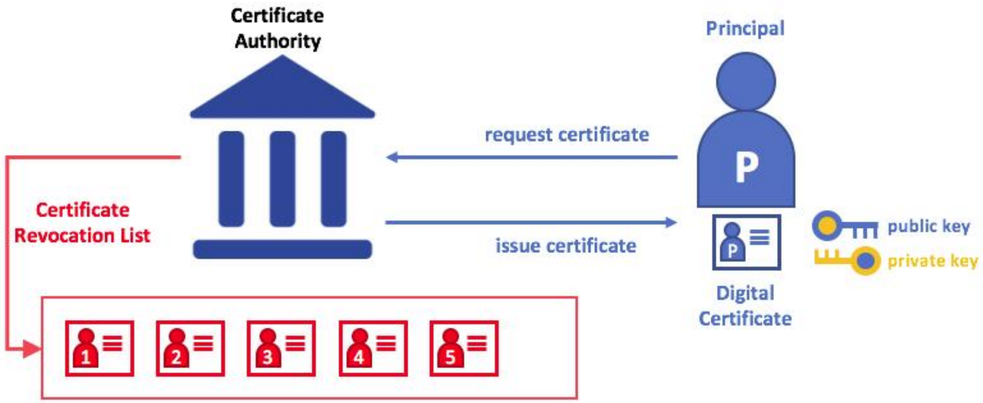
组件性CRDs和功能性CRDs

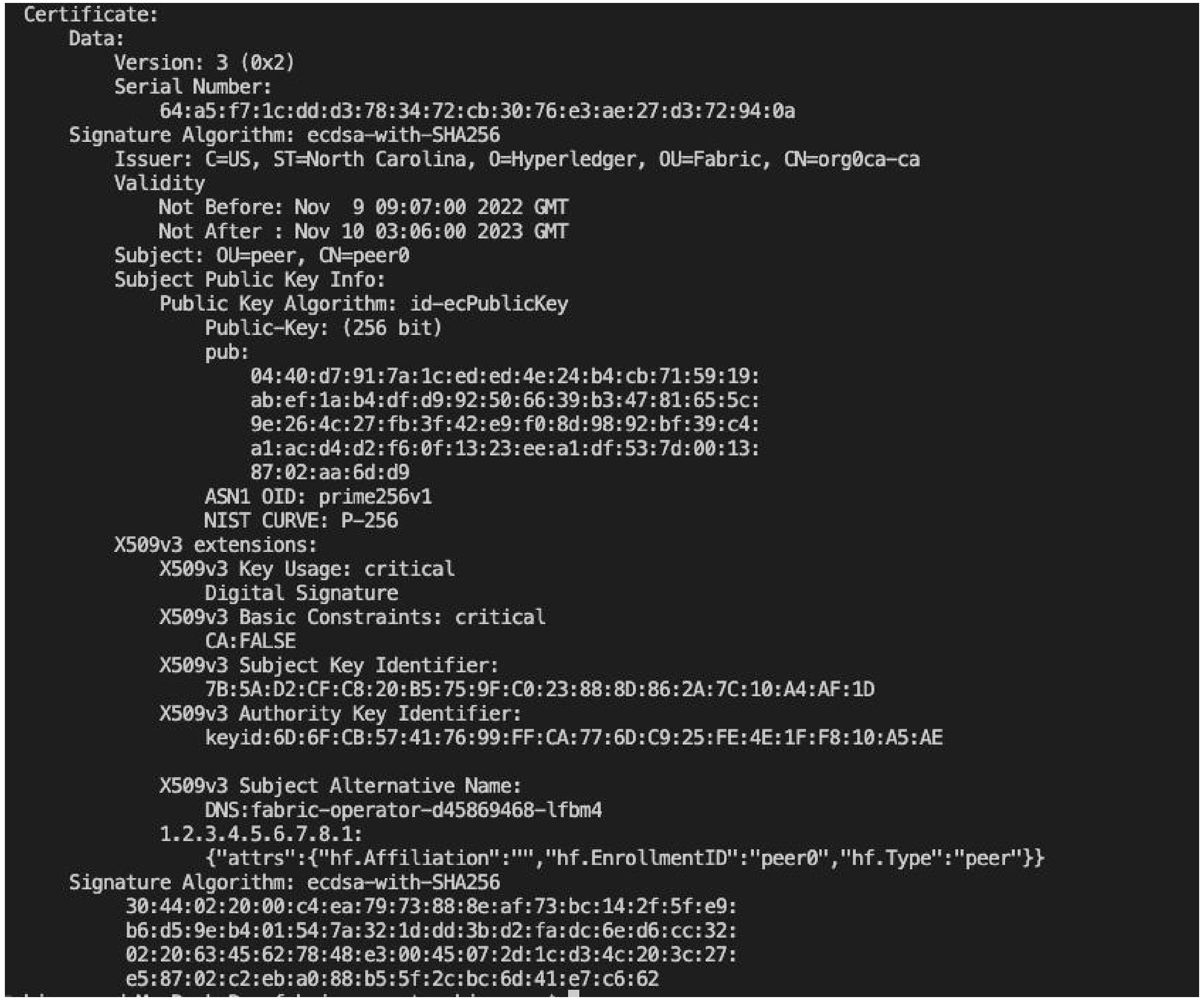
一切皆可CRD...

PKI体系

PKI中的四个关键元素：

- Digital Certificates: x509数字证书
- Public and Private Keys： ECDSA公私钥对
- Certificate Authorities: CA证书授权机构
- Certificate Revocation List: 证书注销列表



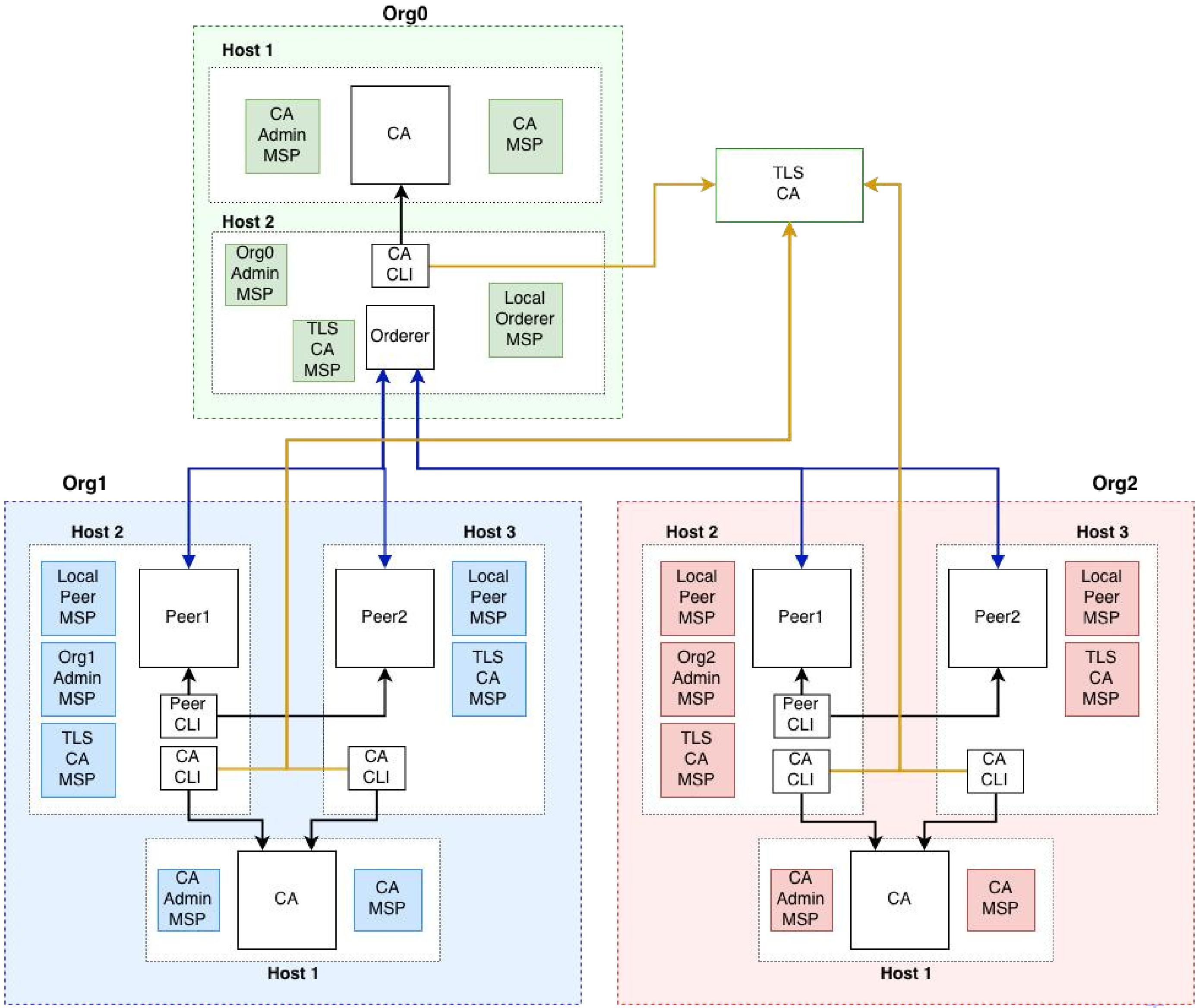


```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            64:a5:f7:1c:dd:d3:78:34:72:cb:30:76:e3:ae:27:d3:72:94:0a
    Signature Algorithm: ecdsa-with-SHA256
        Issuer: C=US, ST=North Carolina, O=Hyperledger, OU=Fabric, CN=org0ca-ca
        Validity
            Not Before: Nov  9 09:07:00 2022 GMT
            Not After : Nov 10 03:06:00 2023 GMT
        Subject: OU=peer, CN=peer0
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:
                    04:40:d7:91:7a:1c:ed:ed:4e:24:b4:cb:71:59:19:
                    ab:ef:1a:b4:df:d9:92:50:66:39:b3:47:81:65:5c:
                    9e:26:4c:27:fb:3f:42:e9:f0:8d:98:92:bf:39:c4:
                    a1:ac:d4:d2:f6:0f:13:23:ee:a1:df:53:7d:00:13:
                    87:02:aa:6d:d9
                ASN1 OID: prime256v1
                NIST CURVE: P-256
        X509v3 extensions:
            X509v3 Key Usage: critical
                Digital Signature
            X509v3 Basic Constraints: critical
                CA:FALSE
            X509v3 Subject Key Identifier:
                7B:5A:D2:CF:C8:20:B5:75:9F:C0:23:88:8D:86:2A:7C:10:A4:AF:1D
            X509v3 Authority Key Identifier:
                keyid:6D:6F:CB:57:41:76:99:FF:CA:77:6D:C9:25:FE:4E:1F:F8:10:A5:AE

            X509v3 Subject Alternative Name:
                DNS:fabric-operator-d45869468-lfbm4
            1.2.3.4.5.6.7.8.1:
                {"attrs":{"hf.Affiliation":"","hf.EnrollmentID":"peer0","hf.Type":"peer"}}
    Signature Algorithm: ecdsa-with-SHA256
         30:44:02:20:00:c4:ea:79:73:88:8e:af:73:bc:14:2f:5f:e9:
         b6:d5:9e:b4:01:54:7a:32:1d:dd:3b:d2:fa:dc:6e:d6:cc:32:
         02:20:63:45:62:78:48:e3:00:45:07:2d:1c:d3:4c:20:3c:27:
         e5:87:02:c2:eb:a0:88:b5:5f:2c:bc:6d:41:e7:c6:62
```

CA实践拓扑

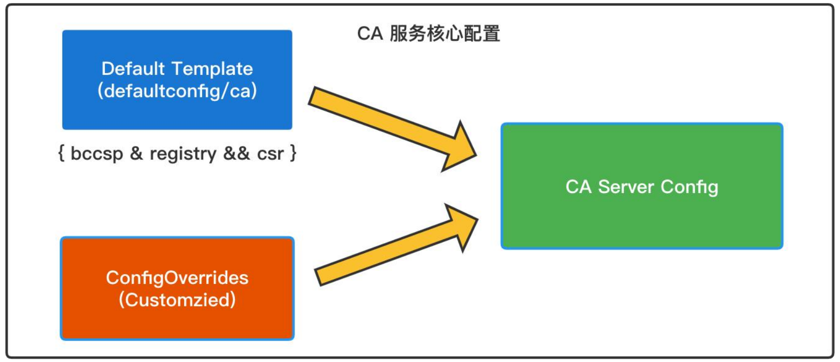
三个机构区块链网络中CA的使用：

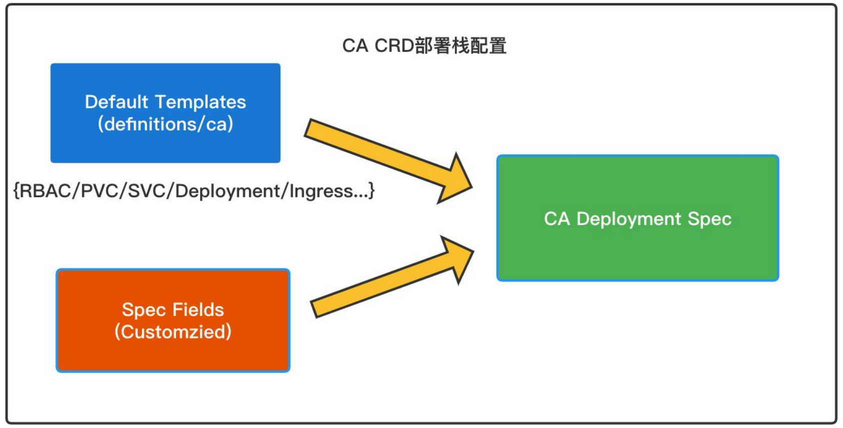
- Org0\Org1\Org2各自部署一个CA
- Org0\Org2\Org3共用一个TLSCA

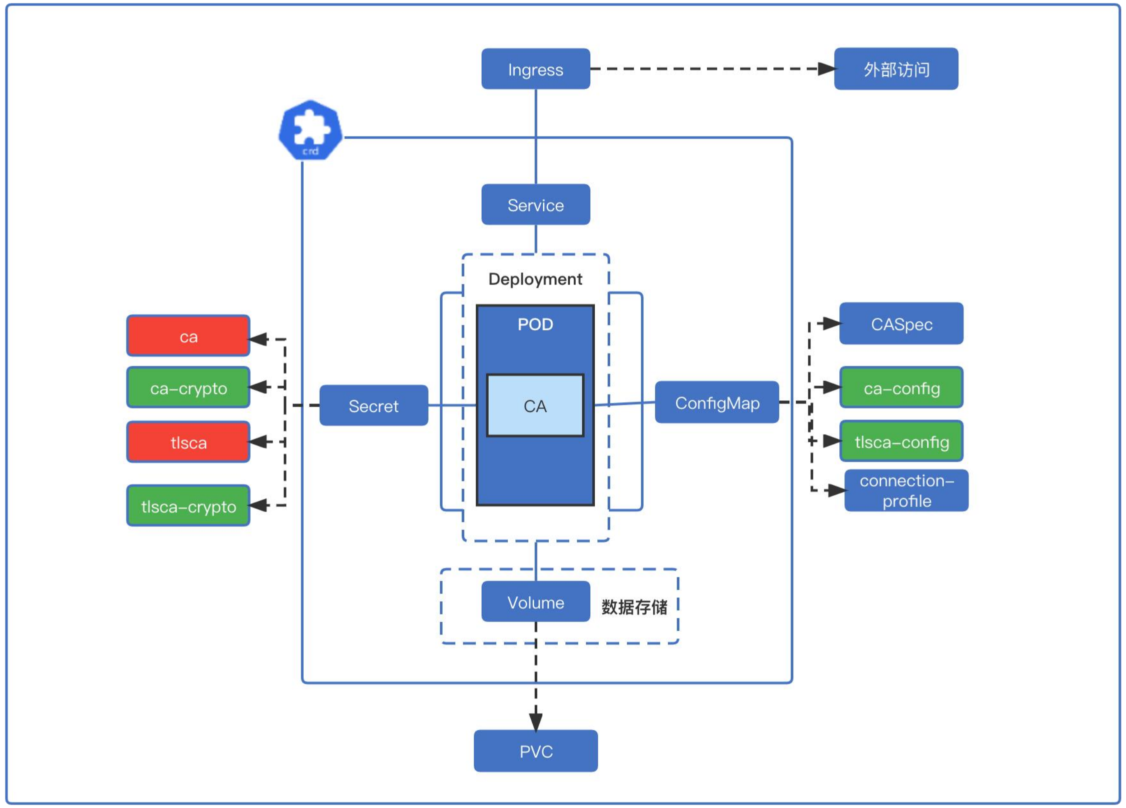


CRD: CA

CA服务提供ECA和TLSCA能力

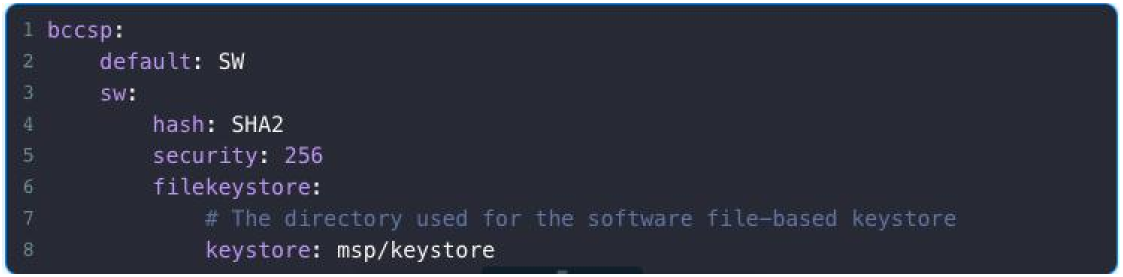




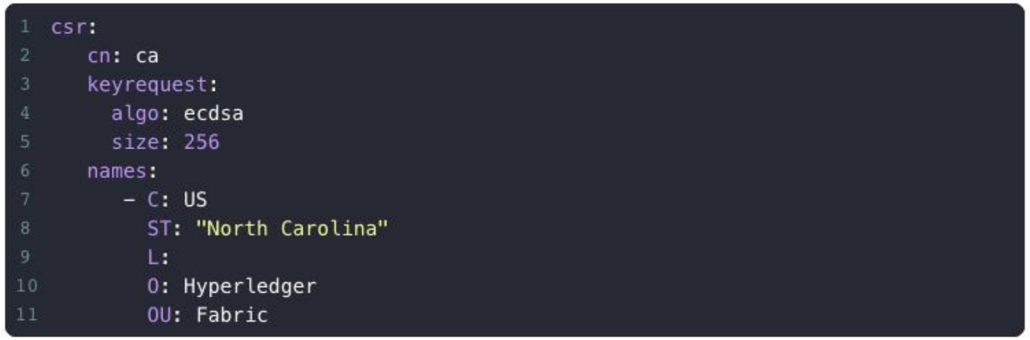


CRD: CA

密码学配置(默认)

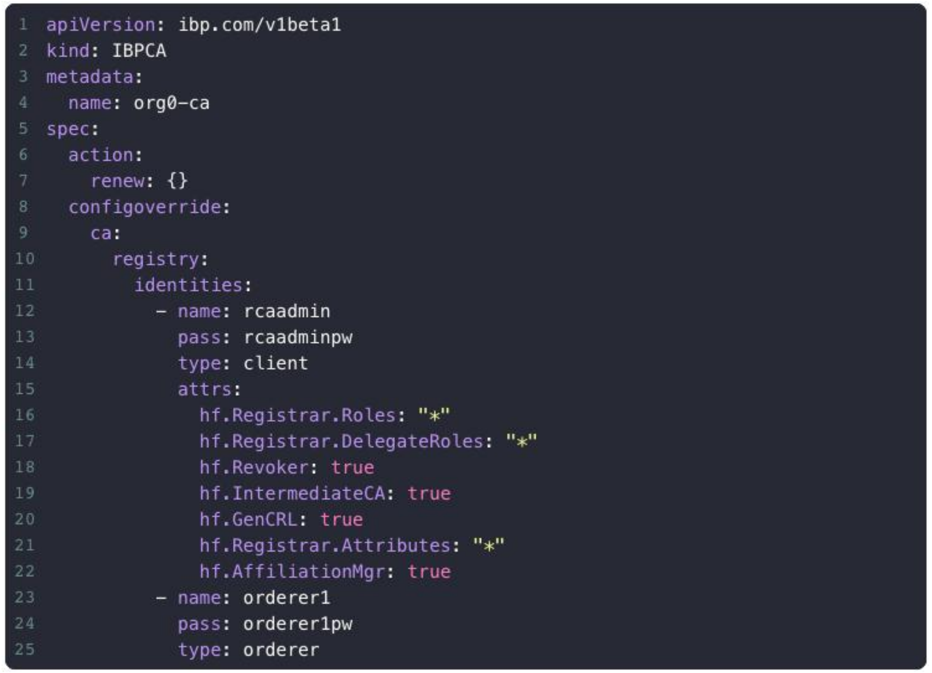


```
bccsp:
    default: SW
    sw:
        hash: SHA2
        security: 256
        filekeystore:
            # The directory used for the software file-based keystore
            keystore: msp/keystore
```

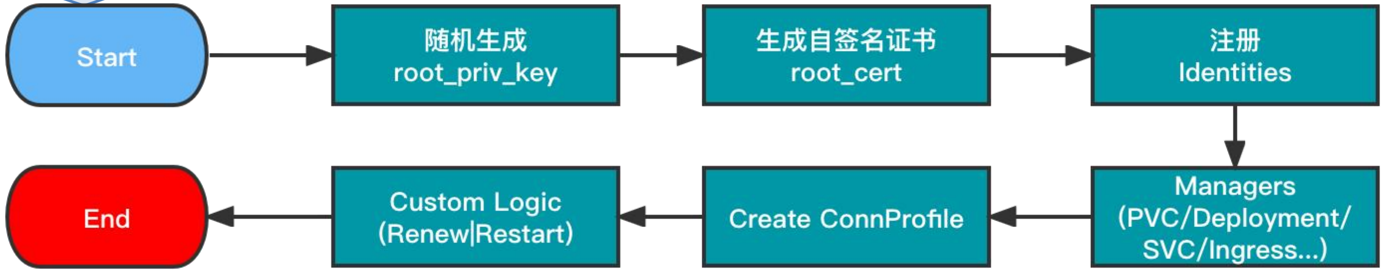


```
csr:
   cn: ca
   keyrequest:
     algo: ecdsa
     size: 256
   names:
      - C: US
        ST: "North Carolina"
        L:
        O: Hyperledger
        OU: Fabric
```

自定义（sample-network/config/cas/org0-ca.yaml）



```
apiVersion: ibp.com/v1beta1
kind: IBPCA
metadata:
  name: org0-ca
spec:
  action:
    renew: {}
  configoverride:
    ca:
      registry:
        identities:
          - name: rcaadmin
            pass: rcaadminpw
            type: client
            attrs:
              hf.Registrar.Roles: "*"
              hf.Registrar.DelegateRoles: "*"
              hf.Revoker: true
              hf.IntermediateCA: true
              hf.GenCRL: true
              hf.Registrar.Attributes: "*"
              hf.AffiliationMgr: true
          - name: orderer1
            pass: orderer1pw
            type: orderer
```



Start

随机生成 root_priv_key

生成自签名证书 root_cert

注册 Identities

Managers (PVC/Deployment/SVC/Ingress...)

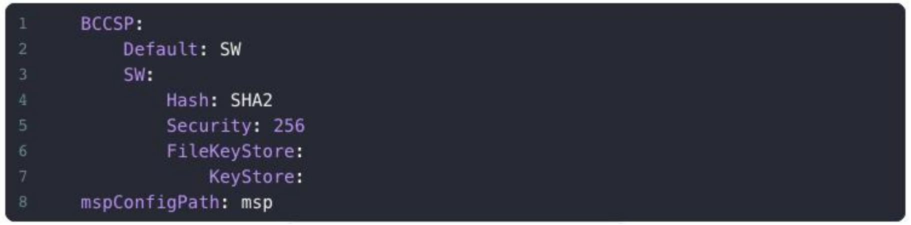
Create ConnProfile

Custom Logic (Renew|Restart)

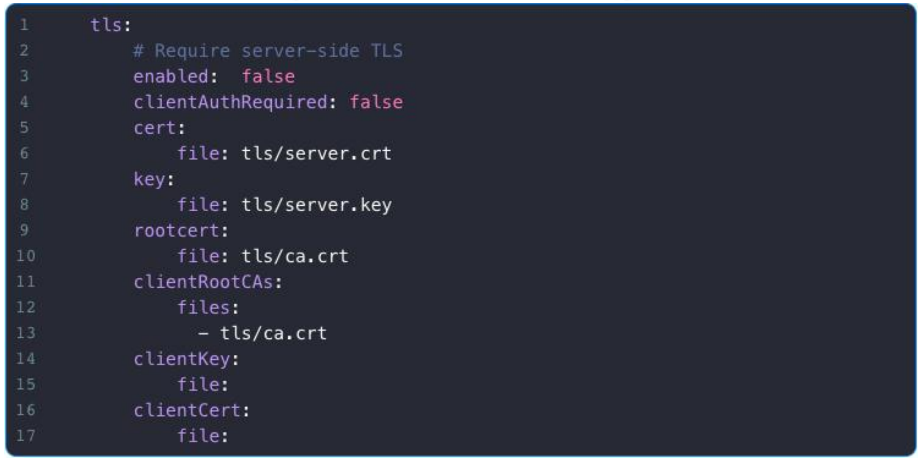
End

Peer账本节点

Peer配置(默认)

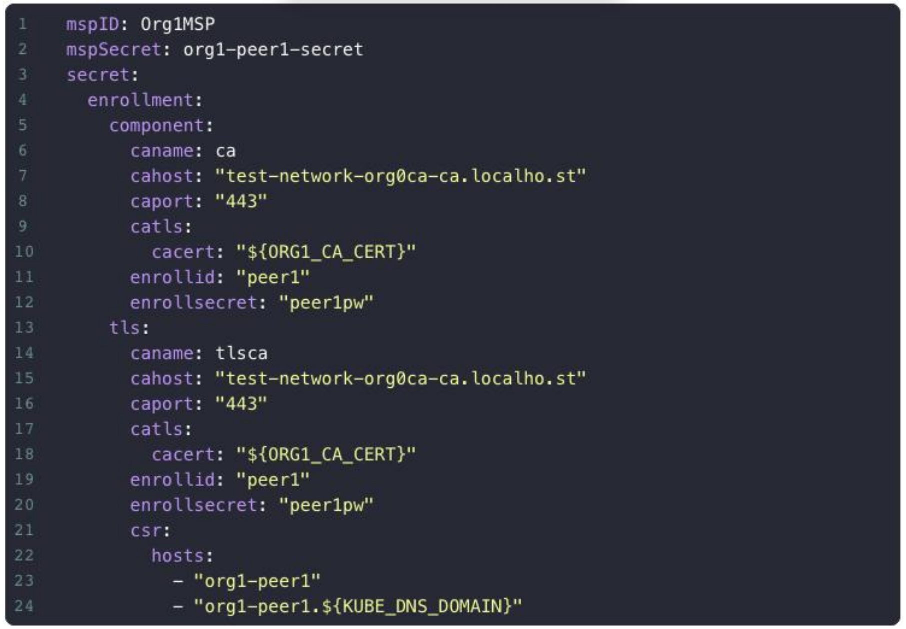


```
BCCSP:
    Default: SW
    SW:
        Hash: SHA2
        Security: 256
        FileKeyStore:
            KeyStore:
mspConfigPath: msp
```

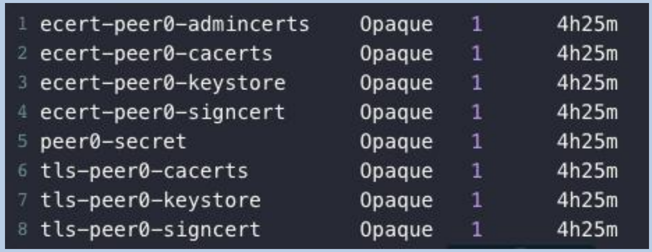


```
tls:
    # Require server-side TLS
    enabled:  false
    clientAuthRequired: false
    cert:
        file: tls/server.crt
    key:
        file: tls/server.key
    rootcert:
        file: tls/ca.crt
    clientRootCAs:
        files:
          - tls/ca.crt
    clientKey:
        file:
    clientCert:
        file:
```

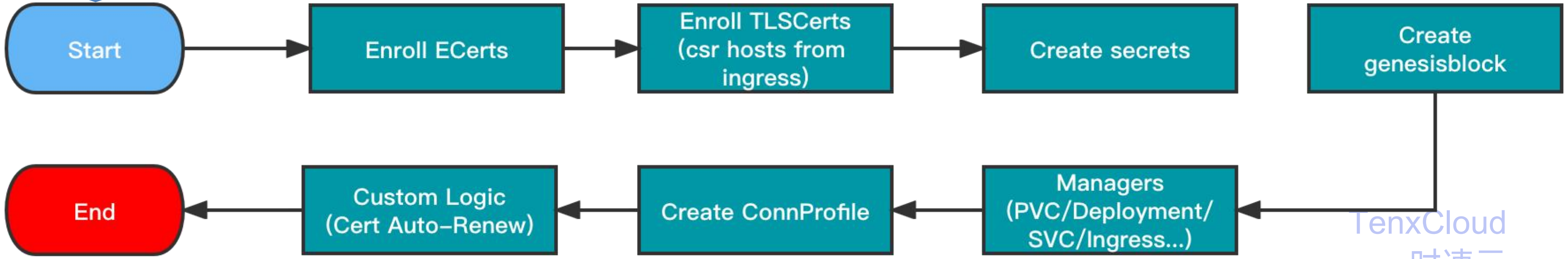
自定义（sample-network/config/peers/org1-peer1.yaml）



```
mspID: Org1MSP
mspSecret: org1-peer1-secret
secret:
  enrollment:
    component:
      caname: ca
      cahost: "test-network-org0ca-ca.localho.st"
      caport: "443"
      catls:
        cacert: "${ORG1_CA_CERT}"
      enrollid: "peer1"
      enrollsecret: "peer1pw"
    tls:
      caname: tlsca
      cahost: "test-network-org0ca-ca.localho.st"
      caport: "443"
      catls:
        cacert: "${ORG1_CA_CERT}"
      enrollid: "peer1"
      enrollsecret: "peer1pw"
      csr:
        hosts:
          - "org1-peer1"
          - "org1-peer1.${KUBE_DNS_DOMAIN}"
```

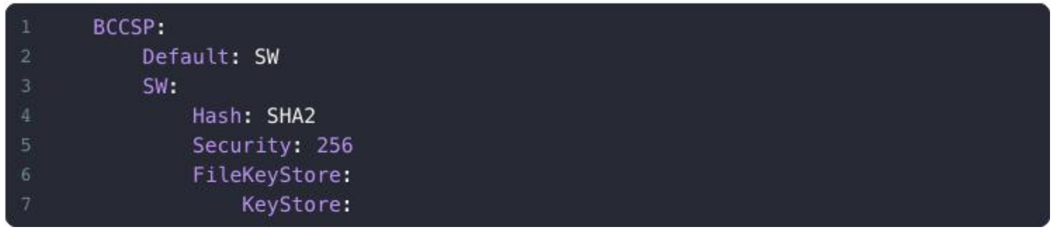


```
ecert-peer0-admincerts    Opaque   1     4h25m
ecert-peer0-cacerts       Opaque   1     4h25m
ecert-peer0-keystore      Opaque   1     4h25m
ecert-peer0-signcert      Opaque   1     4h25m
peer0-secret              Opaque   1     4h25m
tls-peer0-cacerts         Opaque   1     4h25m
tls-peer0-keystore        Opaque   1     4h25m
tls-peer0-signcert        Opaque   1     4h25m
```

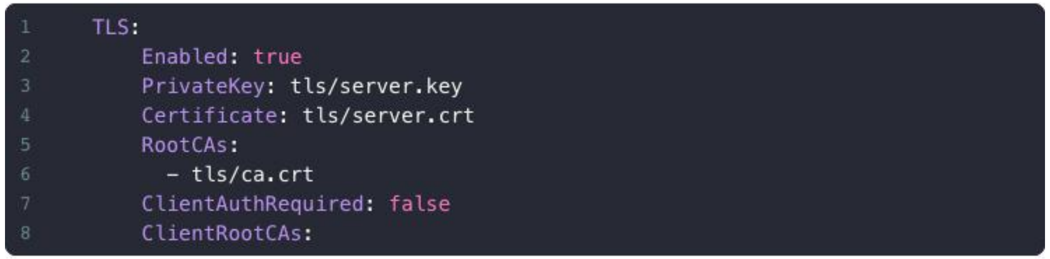

Start
Enroll ECerts
Enroll TLSCerts (csr hosts from ingress)
Create secrets
Create genesisblock
Managers (PVC/Deployment/ SVC/Ingress...)
Create ConnProfile
Custom Logic (Cert Auto-Renew)
End

Orderer排序节点

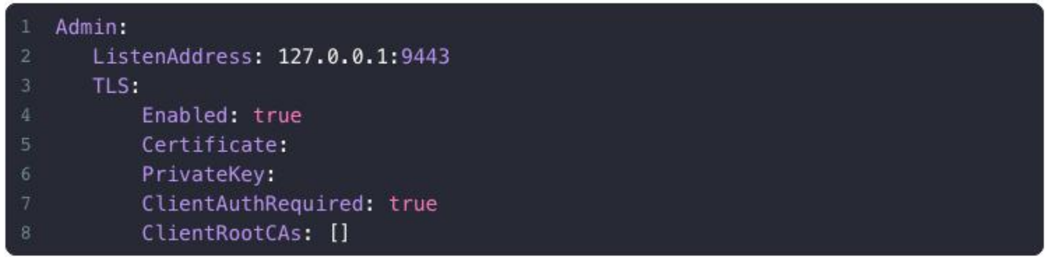
Orderer配置(默认)



```
BCCSP:
    Default: SW
    SW:
        Hash: SHA2
        Security: 256
        FileKeyStore:
            KeyStore:
```

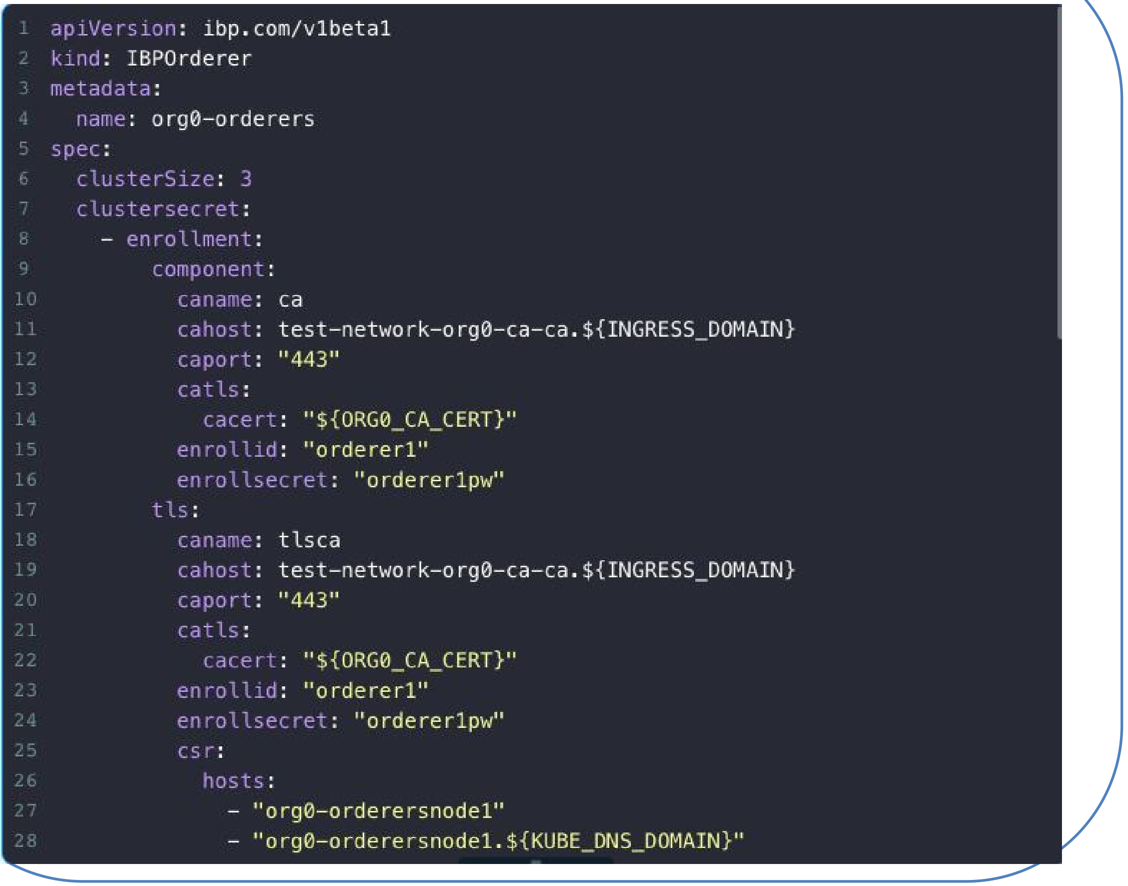


```
TLS:
    Enabled: true
    PrivateKey: tls/server.key
    Certificate: tls/server.crt
    RootCAs:
      - tls/ca.crt
    ClientAuthRequired: false
    ClientRootCAs:
```



```
Admin:
   ListenAddress: 127.0.0.1:9443
   TLS:
       Enabled: true
       Certificate:
       PrivateKey:
       ClientAuthRequired: true
       ClientRootCAs: []
```

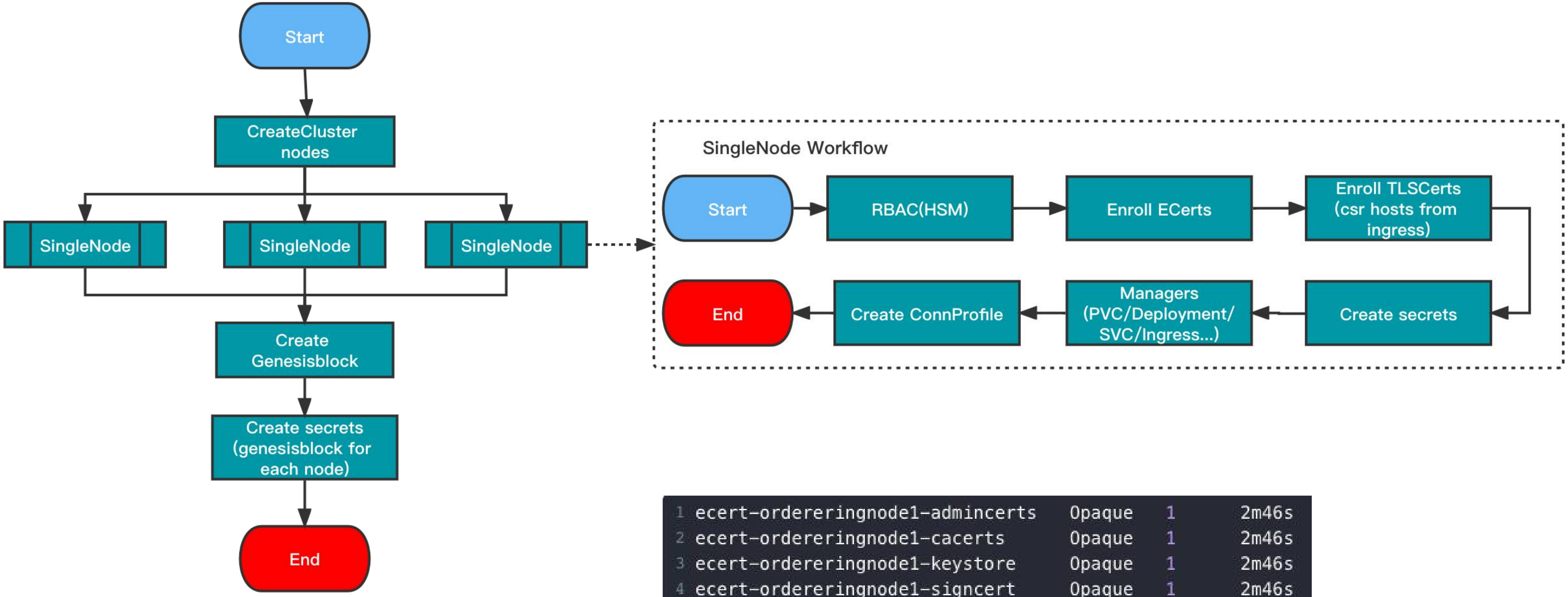
自定义配置 （sample-network/config/orderers/org0-orderers.yaml）

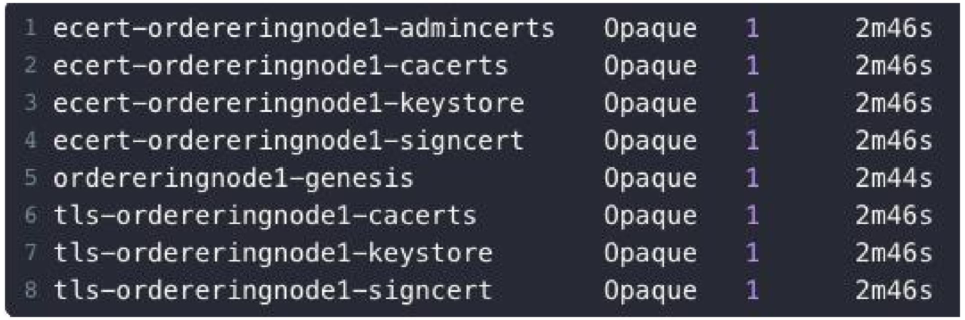


```
apiVersion: ibp.com/v1beta1
kind: IBPOrderer
metadata:
  name: org0-orderers
spec:
  clusterSize: 3
  clustersecret:
    - enrollment:
        component:
          caname: ca
          cahost: test-network-org0-ca-ca.${INGRESS_DOMAIN}
          caport: "443"
          catls:
            cacert: "${ORG0_CA_CERT}"
          enrollid: "orderer1"
          enrollsecret: "orderer1pw"
        tls:
          caname: tlsca
          cahost: test-network-org0-ca-ca.${INGRESS_DOMAIN}
          caport: "443"
          catls:
            cacert: "${ORG0_CA_CERT}"
          enrollid: "orderer1"
          enrollsecret: "orderer1pw"
          csr:
            hosts:
              - "org0-orderersnode1"
              - "org0-orderersnode1.${KUBE_DNS_DOMAIN}"
```

Orderer排序节点

Reconcile流程

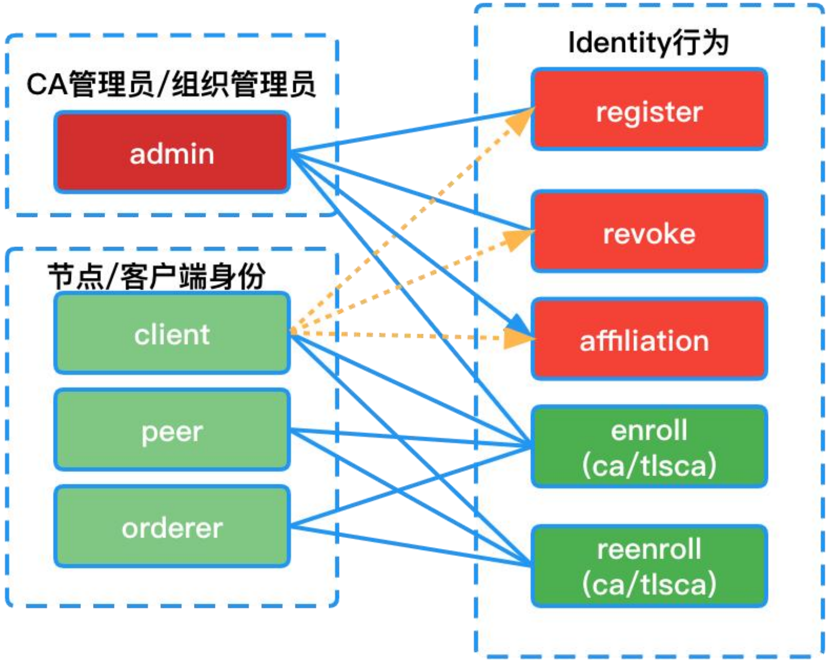


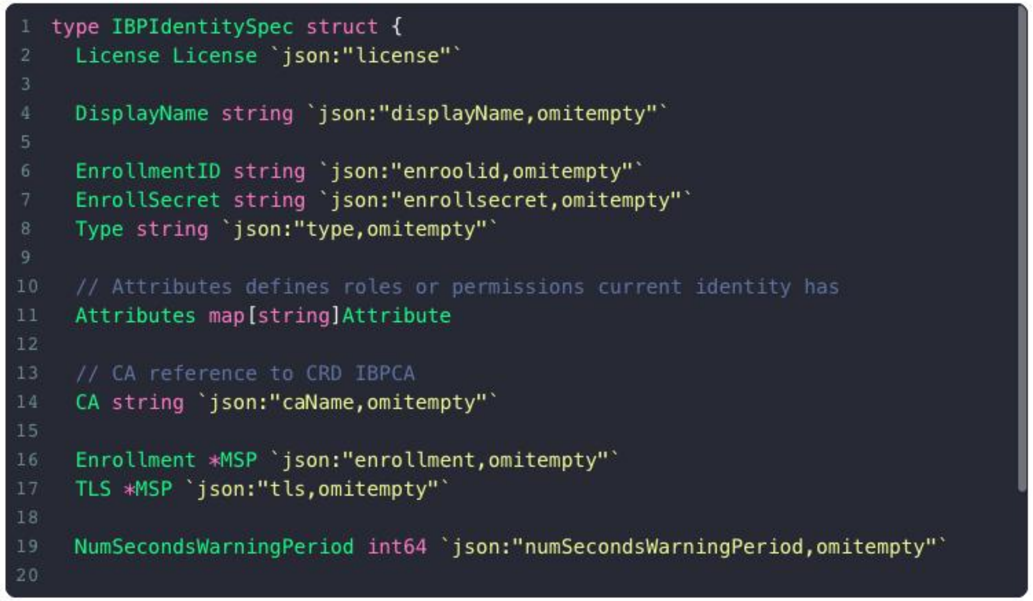


```
ecert-ordereringnode1-admincerts   Opaque   1      2m46s
ecert-ordereringnode1-cacerts      Opaque   1      2m46s
ecert-ordereringnode1-keystore     Opaque   1      2m46s
ecert-ordereringnode1-signcert     Opaque   1      2m46s
ordereringnode1-genesis            Opaque   1      2m44s
tls-ordereringnode1-cacerts        Opaque   1      2m46s
tls-ordereringnode1-keystore       Opaque   1      2m46s
tls-ordereringnode1-signcert       Opaque   1      2m46s
```

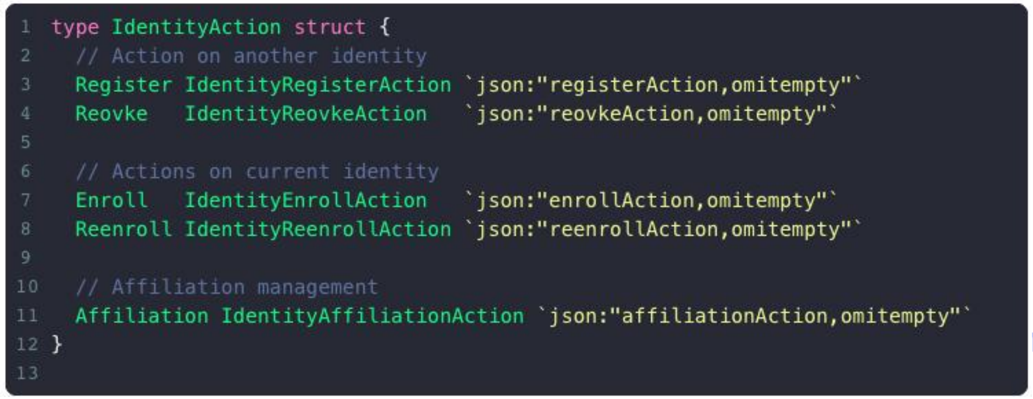
Identity

CRD Identity不是用来代替CA管理所有的身份，而是用来协助BAAS用户托管其拥有的Identity





```go
type IBPIdentitySpec struct {
  License License `json:"license"`

  DisplayName string `json:"displayName,omitempty"`

  EnrollmentID string `json:"enroolid,omitempty"`
  EnrollSecret string `json:"enrollsecret,omitempty"`
  Type string `json:"type,omitempty"`

  // Attributes defines roles or permissions current identity has
  Attributes map[string]Attribute

  // CA reference to CRD IBPCA
  CA string `json:"caName,omitempty"`

  Enrollment *MSP `json:"enrollment,omitempty"`
  TLS *MSP `json:"tls,omitempty"`

  NumSecondsWarningPeriod int64 `json:"numSecondsWarningPeriod,omitempty"`

```



```go
type IdentityAction struct {
  // Action on another identity
  Register IdentityRegisterAction `json:"registerAction,omitempty"`
  Reovke   IdentityReovkeAction   `json:"reovkeAction,omitempty"`

  // Actions on current identity
  Enroll   IdentityEnrollAction   `json:"enrollAction,omitempty"`
  Reenroll IdentityReenrollAction `json:"reenrollAction,omitempty"`

  // Affiliation management
  Affiliation IdentityAffiliationAction `json:"affiliationAction,omitempty"`
}

```

“ **03**

TODO... ”