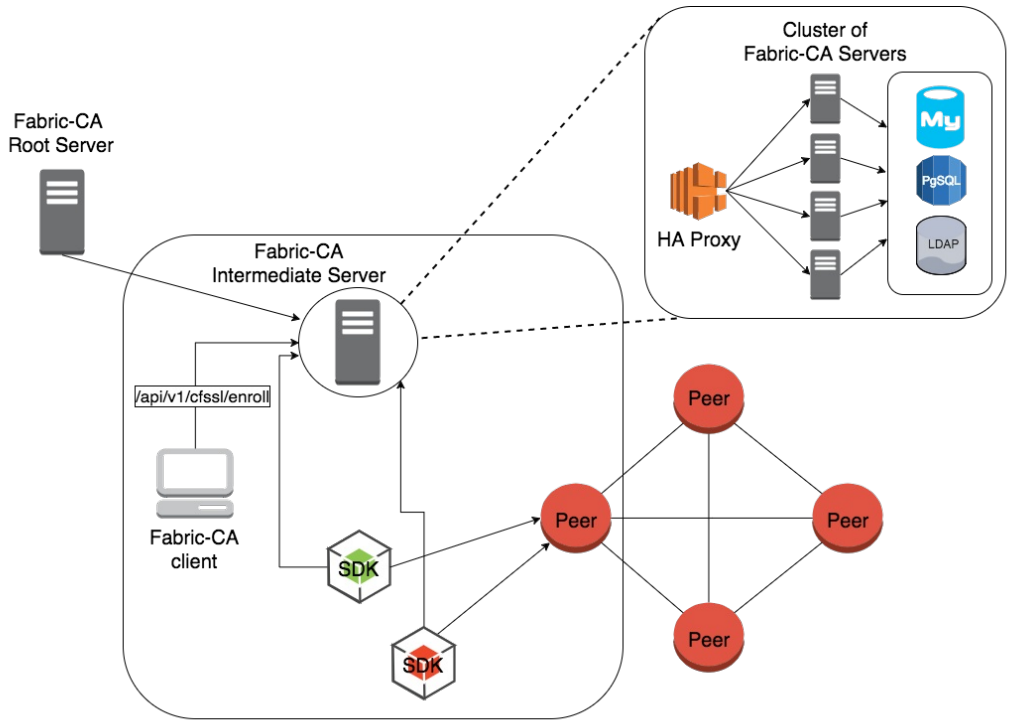


Blockchair

Fabric CA 用户指南

Fabric CA 是 Hyperledger Fabric 的官方配套认证设施。

原文链接：http://hyperledger-fabric.readthedocs.io/en/latest/Setup/ca-setup.html

它提供的功能有：

身份认证，或者从 LDAP 中获取注册信息；

发行担保证书 ECerts (Enrollment Certificates)；

发行交易证书 TCerts (Transaction Certificates)，保障 Hyperledger Fabric 区域链交易平台上的信息匿名性和不可追踪性；

证书更新和撤销。

Fabric CA 属于典型的 CS (Client and Server) 架构，官方代码库：https://github.com/hyperledger/fabric-ca 。

本文目录

登记信息

担保背书（发放证书）

重新背书（更新证书）

撤销证书与清除注册信息

启用 TLS

附录

概述

开篇的图示展现了 Fabric CA 服务端是如何参与到 Hyperledger Fabric 整体架构中去的。

与 Fabric CA 服务端交互的方式有如下两种：

通过 Fabric CA 客户端

使用某种 Fabric SDK

与 Fabric CA 服务端的所有通信，都是通过 REST API 进行的。详情可查看 fabric-ca/swagger/swagger-fabric-ca.json 处的 swagger 文档中的 REST API 部分。

如前图所示，Fabric CA 客户端或 SDK 的请求首先会到达 Fabric CA 集群前端的高可用负载均衡服务端，实际的 CA 服务由后端的某台Fabric CA 服务端提供。同一集群中的所有 Fabric CA 服务端共享相同的后端数据库（或 LDAP）集群，以确保证书和身份的一致性。

入门

运行环境要求

Go 语言 1.7 及以上版本

已正确设置 **GOPATH** 环境变量

已安装 libtool 与 libtdhl-dev 包（更多信息请参看：https://www.gnu.org/software/libtool/）

安装

如下所示，将安装 fabric-ca-server 与 fabric-ca-client 两个终端命令行工具。

```
# go get -u github.com/hyperledger/fabric-ca/cmd/...
```

启动服务端：本地环境

如下所示，将按默认配置启动 Fabric CA 服务端，-b 选项用于指定管理员的账号与密码。

```
# fabric-ca-server start -b admin:adminpw
```

默认将在当前目录创建一个名为 fabric-ca-server-config.yaml 的配置文件，该文件的存储位置也可以另行指定。

启动服务端：Docker 环境

也可以选择在 Docker 环境下运行服务端，如下将创建并通过 docker-compose 启动服务端

```
# cd $GOPATH/src/github.com/hyperledger/fabric-ca
# make docker
# cd docker/server
# docker-compose up -d
```

名为 hyperledger/fabric-ca 的 docker 镜像中包含了 fabric-ca-server 与 fabric-ca-client 命令行工具。

Fabric CA 终端命令概览

fabric-ca-server 命令用法：

```
Hyperledger Fabric Certificate Authority Server

Usage:
  fabric-ca-server [command]

Available Commands:
  init        Initialize the Fabric CA server
  start       Start the Fabric CA server

Flags:
      --address string                        Listening address of Fabric CA server (default "0.0.0.0")
  -b, --boot string                           The user:pass for bootstrap admin which is required to build default config file
      --ca.certfile string                    PEM-encoded CA certificate file (default "ca-cert.pem")
      --ca.chainfile string                   PEM-encoded CA chain file (default "ca-chain.pem")
      --ca.keyfile string                     PEM-encoded CA key file (default "ca-key.pem")
  -n, --ca.name string                        Certificate Authority name
  -c, --config string                         Configuration file (default "fabric-ca-server-config.yaml")
      --csr.cn string                         The common name field of the certificate signing request to a parent Fabric CA server
      --csr.hosts stringSlice                 A list of space-separated host names in a certificate signing request to a parent Fabric CA server
      --csr.serialnumber string               The serial number in a certificate signing request to a parent Fabric CA server
      --db.datasource string                  Data source which is database specific (default "fabric-ca-server.db")
      --db.tls.certfiles stringSlice          PEM-encoded list of trusted
```

```
certificate files
      --db.tls.client.certfile string          PEM-encoded certificate file
when mutual authenticate is enabled
      --db.tls.client.keyfile string           PEM-encoded key file when
mutual authentication is enabled
      --db.type string                         Type of database; one of:
sqlite3, postgres, mysql (default "sqlite3")
  -d, --debug                                  Enable debug level logging
      --ldap.enabled                           Enable the LDAP client for
authentication and attributes
      --ldap.groupfilter string                The LDAP group filter for a
single affiliation group (default "(memberUid=%s)")
      --ldap.url string                        LDAP client URL of form ldap://
adminDN:adminPassword@host[:port]/base
      --ldap.userfilter string                 The LDAP user filter to use
when searching for users (default "(uid=%s)")
  -p, --port int                               Listening port of Fabric CA
server (default 7054)
      --registry.maxenrollments int            Maximum number of enrollments;
valid if LDAP not enabled
      --tls.certfile string                    PEM-encoded TLS certificate
file for server's listening port (default "ca-cert.pem")
      --tls.clientauth.certfiles stringSlice   PEM-encoded list of trusted
certificate files
      --tls.clientauth.type string             Policy the server will follow
for TLS Client Authentication. (default "noclientcert")
      --tls.enabled                            Enable TLS on the listening
port
      --tls.keyfile string                     PEM-encoded TLS key for
server's listening port (default "ca-key.pem")
  -u, --url string                             URL of the parent Fabric CA
server


Use "fabric-ca-server [command] --help" for more information about a command.
```

fabric-ca-client 命令用法：

```
# fabric-ca-client
Hyperledger Fabric Certificate Authority Client

Usage:
  fabric-ca-client [command]

Available Commands:
  enroll      Enroll an identity
  getcacert   Get CA certificate chain
  reenroll    Reenroll an identity
  register    Register an identity
  revoke      Revoke an identity

Flags:
  -c, --config string               Configuration file (default
"$HOME/.fabric-ca-client/fabric-ca-client-config.yaml")
      --csr.cn string               The common name field of the certificate
signing request
      --csr.hosts stringSlice       A list of space-separated host names in a
certificate signing request
      --csr.serialnumber string     The serial number in a certificate
```

```
signing request
19    -d, --debug                          Enable debug level logging
20        --enrollment.hosts string        Comma-separated host list
21        --enrollment.label string        Label to use in HSM operations
22        --enrollment.profile string      Name of the signing profile to use in
issuing the certificate
23        --id.affiliation string          The identity's affiliation
24        --id.attr string                 Attributes associated with this identity
(e.g. hf.Revoker=true)
25        --id.maxenrollments int          The maximum number of times the secret
can be reused to enroll
26        --id.name string                 Unique name of the identity
27        --id.secret string               The enrollment secret for the identity
being registered
28        --id.type string                 Type of identity being registered (e.g.
'peer, app, user')
29    -M, --mspdir string                  Membership Service Provider directory
(default "msp")
30    -m, --myhost string                  Hostname to include in the certificate
signing request during enrollment (default "$HOSTNAME")
31        --tls.certfiles stringSlice      PEM-encoded list of trusted certificate
files
32        --tls.client.certfile string     PEM-encoded certificate file when mutual
authenticate is enabled
33        --tls.client.keyfile string      PEM-encoded key file when mutual
authentication is enabled
34    -u, --url string                     URL of the Fabric CA server (default
"http://localhost:7054")
35
36 Use "fabric-ca-client [command] --help" for more information about a command.
```

注：参数类型标记为“stringSlice”的选项，表示可以批量指定多个参数，形如——"string0 string1 ... stringN" 的形式，此时外层必须有双引号，且各项之间以空格分开；多次分开指定则不需要加双引号，如 -csr.hosts "host1 host2" 与 -csr.hosts host1 -csr.hosts host2 效果相同。

配置文件格式

服务端配置文件格式

服务端启动时，可以通过 -c 或 --config 选项指定配置文件，若目标文件不存在，将在指定路径创建一个默认配置文件（若不提供 -c 或 --config 选项，则在服务端的家目录下创建），内容类似如下：

```
# Server's listening port (default: 7054)
port: 7054

# Enables debug logging (default: false)
debug: false


#############################################################################
#   TLS section for the server's listening port
```

```
 9
############################################################################
10 tls:
11   # Enable TLS (default: false)
12   enabled: false
13   certfile: ca-cert.pem
14   keyfile: ca-key.pem
15
16
############################################################################
17 #  The CA section contains the key and certificate files used when
18 #  issuing enrollment certificates (ECerts) and transaction
19 #  certificates (TCerts).
20
############################################################################
21 ca:
22   # Certificate file (default: ca-cert.pem)
23   certfile: ca-cert.pem
24   # Key file (default: ca-key.pem)
25   keyfile: ca-key.pem
26
27
############################################################################
28 #  The registry section controls how the Fabric CA server does two
things:
29 #  1) authenticates enrollment requests which contain identity name and
30 #     password (also known as enrollment ID and secret).
31 #  2) once authenticated, retrieves the identity's attribute names and
32 #     values which the Fabric CA server optionally puts into TCerts
33 #     which it issues for transacting on the Hyperledger Fabric
blockchain.
34 #     These attributes are useful for making access control decisions in
35 #     chaincode.
36 #  There are two main configuration options:
37 #  1) The Fabric CA server is the registry
38 #  2) An LDAP server is the registry, in which case the Fabric CA server
39 #     calls the LDAP server to perform these tasks.
40
############################################################################
41 registry:
42   # Maximum number of times a password/secret can be reused for
enrollment
43   # (default: 0, which means there is no limit)
44   maxEnrollments: 0
45
46   # Contains identity information which is used when LDAP is disabled
47   identities:
48      - name: <<<ADMIN>>>
49        pass: <<<ADMINPW>>>
50        type: client
51        affiliation: ""
52        attrs:
53           hf.Registrar.Roles: "client,user,peer,validator,auditor,ca"
54           hf.Registrar.DelegateRoles: "client,user,validator,auditor"
55           hf.Revoker: true
56           hf.IntermediateCA: true
57
58
############################################################################
59 #  Database section
60 #  Supported types are: "sqlite3", "postgres", and "mysql".
61 #  The datasource value depends on the type.
62 #  If the type is "sqlite3", the datasource value is a file name to use
```

```
 63 #  as the database store.  Since "sqlite3" is an embedded database, it
 64 #  may not be used if you want to run the Fabric CA server in a cluster.
 65 #  To run the Fabric CA server in a cluster, you must choose "postgres"
 66 #  or "mysql".
 67
###############################################################################
 68 db:
 69   type: sqlite3
 70   datasource: fabric-ca-server.db
 71   tls:
 72       enabled: false
 73       certfiles:
 74         - db-server-cert.pem
 75       client:
 76         certfile: db-client-cert.pem
 77         keyfile: db-client-key.pem
 78
 79
###############################################################################
 80 #  LDAP section
 81 #  If LDAP is enabled, the Fabric CA server calls LDAP to:
 82 #  1) authenticate enrollment ID and secret (i.e. identity name and
password)
 83 #     for enrollment requests
 84 #  2) To retrieve identity attributes
 85
###############################################################################
 86 ldap:
 87    # Enables or disables the LDAP client (default: false)
 88    enabled: false
 89    # The URL of the LDAP server
 90    url: ldap://<adminDN>:<adminPassword>@<host>:<port>/<base>
 91    tls:
 92       certfiles:
 93         - ldap-server-cert.pem
 94       client:
 95          certfile: ldap-client-cert.pem
 96          keyfile: ldap-client-key.pem
 97
 98
###############################################################################
 99 #  Affiliation section
100
###############################################################################
101 affiliations:
102    org1:
103       - department1
104       - department2
105    org2:
106       - department1
107
108
###############################################################################
109 #  Signing section
110
###############################################################################
111 signing:
112     profiles:
113       ca:
114          usage:
115            - cert sign
116          expiry: 8000h
117          caconstraint:
```

```
            isca: true
     default:
       usage:
         - cert sign
       expiry: 8000h


###############################################################################
#  Certificate Signing Request section for generating the CA certificate
###############################################################################
csr:
   cn: fabric-ca-server
   names:
      - C: US
        ST: North Carolina
        L:
        O: Hyperledger
        OU: Fabric
   hosts:
     - <<<MYHOST>>>
   ca:
      pathlen:
      pathlenzero:
      expiry:


###############################################################################
#  Crypto section configures the crypto primitives used for all
###############################################################################
crypto:
  software:
     hash_family: SHA2
     security_level: 256
     ephemeral: false
     key_store_dir: keys
```

客户端配置文件格式

客户端启动时，可以通过 -c 或 --config 选项指定配置文件，若目标文件不存在，将在指定路径创建一个默认配置文件（若不提供 -c 或 --config 选项，则在客户端的家目录下创建），内容类似如下：

```
###############################################################################
# Client Configuration
###############################################################################

# URL of the Fabric CA server (default: http://localhost:7054)
URL: http://localhost:7054

# Membership Service Provider (MSP) directory
# When the client is used to enroll a peer or an orderer, this field must be
# set to the MSP directory of the peer/orderer
MSPDir:

```

```
###############################################################################
#    TLS section for secure socket connection
###############################################################################
tls:
  # Enable TLS (default: false)
  enabled: false
  certfiles:
  client:
    certfile:
    keyfile:

###############################################################################
#  Certificate Signing Request section for generating the CSR for
#  an enrollment certificate (ECert)
###############################################################################
csr:
  cn: <<<ENROLLMENT_ID>>>
  names:
    - C: US
      ST: North Carolina
      L:
      O: Hyperledger
      OU: Fabric
  hosts:
   - <<<MYHOST>>>
  ca:
    pathlen:
    pathlenzero:
    expiry:

###############################################################################
#  Registration section used to register a new identity with Fabric CA server
###############################################################################
id:
  name:
  type:
  affiliation:
  attributes:
    - name:
      value:

###############################################################################
#  Enrollment section used to enroll an identity with Fabric CA server
###############################################################################
enrollment:
  hosts:
  profile:
  label:
```

配置信息的优先级

如下三种方式，优先级依次降低（即：命令行优先于环境变量，环境变量优先于配置文件）：

通过命令行选项指定

通过设置环境变量指定

写入配置文件

如下所示，配置文件中指定了证书名称：

```
tls:
  # Enable TLS (default: false)
  enabled: false

  # TLS for the client's listenting port (default: false)
  certfiles:
  client:
    certfile: cert.pem
    keyfile:
```

若之后再定义如下环境变量，则有效证书名称为：cert2.pem

```
export FABRIC_CA_CLIENT_TLS_CLIENT_CERTFILE=cert2.pem
```

若之后再通过命令行指定如下内容，则有效证书名称为：cert3.pem

```
fabric-ca-client enroll --tls.client.certfile cert3.pem
```

注：配置文件中指定文件路径时，可以使用相对路径（相对于该配置文件所在位置）或绝对路径。

Fabric CA 服务端

这一部分将详细描述Fabric CA 服务端。

服务端家目录位置，按如下规则确定：

若已设定环境变量 **FABRIC_CA_SERVER_HOME**，以此为准

否则，若已设定 **FABRIC_CA_HOME**，以此为准

否则，若已设定 **CA_CFG_PATH**，以此为准

否则，使用当前目录，即：**$PWD**

本部分接下来的介绍，将假设 **FABRIC_CA_HOME** 已初设定为 **$HOME/fabric-ca/server** ，且服务端配置文件位于此目录下。

初始化服务端

如前所述，初始化服务端使用如下形式：

```
# fabric-ca-server init -b admin:adminpw
```

初始化时，需要 -b (bootstrap identity) 选项；服务端启动，需要至少有一个自我认证的身份存在。

服务端配置文件中有一个 CSR (Certificate Signing Request) 区域，如下是一个示例。

如果需果通过 TLS 远程连接到服务端，请将 cn 字段的值替换为服务端 IP 或 域名：

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

以上字段是由 fabric-ca-init 生成的，用于 X.509 签名密钥和证书，作用于配置文件中指定的 ca.certfile 与 ca.keyfile 。

各字段含义如下：

cn：Common Name

key：指定密钥算法和密钥长度

O：organization name

OU：organization unit

L：location or city

ST：state

C：country

如果 CSR 区域的值需要自定义，首先删除 ca.certfile 与 ca.keyfile 字段指定的文件，然后再次运行 fabric-ca-server init -b ***admin:adminpw*** 。

如果指定了 -u <parent-fabric-ca-server-URL> 选项，则 Fabric CA 服务端自身的证书由上级 CA 签发；否则，fabric-ca-server init 将生成一个自签证书，同时在服务端家目录下生成一个名为 fabric-ca-server-config.yaml 的默认配置文件。

算法和密钥长度：

在 CSR 中可以指定用于生成 X.509 密钥和证书的算法，可选 RSA 或 ECDSA (Elliptic Curve Digital Signature Algorithm)；如下示例使用 ecdsa-with-SHA256 算法：

```
key:
   algo: ecdsa
   size: 256
```

ECDSA 可选的密钥长度：256、384、512；**RSA** 可选的密钥长度：2048、4096 。

启动服务端

如下形式将启动 Fabric CA 服务端：

```
# fabric-ca-server start -b admin:adminpw
```

如果此前没有执行 fabric-ca-server init，将首先执行初始化动作，生成 ca-cert.pem、ca-key.pem 及一个默认配置文件。

除非使用基于 LDAP 的用户认证，否则需要至少一个自认证管理员身份，用于登记和认证其它身份；此处的 -b 选项与 init 时的意义相同。

如果需要限制同一个管理员密码可以发放的证书总量，可以设置 registry.maxEnrollments 为适当的值，若设置为 0，表示无限制，默认为 0。

Fabric CA 服务端默认监听在 7054 端口。

配置数据库

Fabric CA 默认使用的数据库是 SQLite，默认数据库文件位于服务端家目录下的 fabric-ca-server.db。

可以选择使用 PostgreSQL 或 MySQL。

PostgreSQL

官方手册：https://www.postgresql.org/docs/manuals

如下示例配置可用于连接 PostgreSQL，请确保其中的各个变量值被正确设置。

```
db:
  type: postgres
  datasource: host=localhost port=5432 user=Username password=Password
dbname=fabric-ca-server sslmode=verify-full
```

如果需要使用 TLS 连接数据库，则必须设置 Fabric CA 服务端配置文件中的 db.tls 部分；若 PostgreSQL 服务端启用了针对客户端的 SSL 认证，则需要同时指定 db.tls.client 部分。

如下是一份 db.tls 配置示例：

```
db:
  ...
  tls:
      enabled: true
      certfiles:
        - db-server-cert.pem
      client:
            certfile: db-client-cert.pem
            keyfile: db-client-key.pem
```

certfiles 字段用于指定一个或多个 PEM 格式编码的可信 ROOT CA 证书文件；certfile 与 keyfile 指定 PEM 格式编码的证书和私钥，在与 PostgreSQL 服务端通信时，用于证明 Fabric CA 服务端的合法身份。

MySQL

略...

配置 LDAP

略...

配置集群

HAProxy 配置...略...

Fabric CA 客户端

Fabric CA 客户端的家目录，由如下其中一个条件决定（优先级由高到低）：

FABRIC_CA_CLIENT_HOME 环境变量

FABRIC_CA_HOME 环境变量

CA_CFG_PATH 环境变量

$HOME/.fabric-ca-client 目录

接下来的介绍，假定客户端的配置文件存放于客户端的有效家目录下。

管理员身份自认证

首先，修改配置文件，其中 csr.cn 字段必须与自认证的身份名称相同。CSR 部分默认值如下：

```
csr:
  cn: <<enrollment ID>>
  key:
    algo: ecdsa
    size: 256
  names:
    - C: US
      ST: North Carolina
      L:
      O: Hyperledger Fabric
      OU: Fabric CA
  hosts:
   - <<hostname of the fabric-ca-client>>
  ca:
    pathlen:
    pathlenzero:
    expiry:
```

之后，执行 fabric-ca-client enroll 命令进行身份认证。

如下示例，通过调用监听在本地 7054 端口的服务端，对 ID：admin 与 PWD：adminpw

的管理员身份进行了自认证：

```
# export FABRIC_CA_CLIENT_HOME=$HOME/fabric-ca/clients/admin
# fabric-ca-client enroll -u http://admin:adminpw@localhost:7054
```

enroll 命令会将生成的 ECert（enrollment certificate），以及对应的私钥及、CA 证书链（PEM 格式）存储在子目录 msp 中，命令执行成功后提示这些文件的存储位置。

登记普通角色身份

执行登记（register）行为的角色本身首先要获得认证，并拥有认证目标角色的权限。

Fabric CA 服务端在执行登记的过程中，会做两项权限检查：

> 执行认证行为的角色必须具有 "hf.Registar.Roles" 属性（此属性的各个值以逗号分隔），并且被认证的身份必须包含在其中。
>
> > 例如：登记角色的 "hf.Registar.Roles" 值是 "peer,app,user"，则它可以登记 peer、app、user 三种角色，但不能登记 orderer 角色
>
> 执行认证行为的角色的组织关系（affiliation）必须与被认证的角色在同一部门或是其上级部门。
>
> > 例如：组织关系为 "a.b" 的登记身份，有权登记 "a.b.c" 部门的角色，但无取登记“a.d” 部门的角色

如下示例，使用 admin 的身份及其配套证书，登记了一个名称为 "admin2"、类型为 "user"、组织关系为 "org1.department1"、"hf.Revoker" 属性为 "true" 的新角色：

```
# export FABRIC_CA_CLIENT_HOME=$HOME/fabric-ca/clients/admin
# fabric-ca-client register --id.name admin2 --id.type user --id.affiliation
org1.department1 --id.attr hf.Revoker=true
```

下一步（认证／enroll）所必须的密码将会被打印出来，其它具有认证权限的身份（如：除 "admin" 之外的管理员）可以使用这个密码，对 "admin2" 进行直接认证，不必是自己先前亲自登记的。

fabric-ca-client 命令的各个选项，都可以预先在配置文件中设置默认值，这样执行对应操作的时候就可以简化选项，若有如下配置：

```
id:
  name:
  type: user
  affiliation: org1.department1
  attributes:
    - name: hf.Revoker
      value: true
    - name: anotherAttrName
      value: anotherAttrValue
```

则上述登记命令可简化为：

```
# export FABRIC_CA_CLIENT_HOME=$HOME/fabric-ca/clients/admin
# fabric-ca-client register --id.name admin2
```

如下命令登记了一个名为 "peer1" 的身份，它是接下来的用于示例的被认证对象：

```
# export FABRIC_CA_CLIENT_HOME=$HOME/fabric-ca/clients/admin
# fabric-ca-client register --id.name peer1 --id.type peer --id.affiliation
org1.department1 --id.secret peer1pw
```

注意，下一步将要用到的认证密码是可以通过 --id.secret 手动指定的。

认证普通角色身份

通过上一步已经成功登记了一个节点的身份，现在可以通过指定被认证角色的 ID 和对应的密码来执行认证，这与管理员角色自认证过程类似，除了此处演示了使用 -M 选项指定 MSP(Membership Service Provider) 路径。

如下命令对 peer1 身份进行了认证，请将 -M 选项的值替换为你所在的 Fabric CA 客户端中名为 core.yaml 的配置文件中的 'mspConfigPath' 字段的值：

```
# export FABRIC_CA_CLIENT_HOME=$HOME/fabric-ca/clients/peer1
# fabric-ca-client enroll -u http://peer1:peer1pw@localhost:7054 -M
$FABRIC_CA_CLIENT_HOME/msp
```

认证 orderer 角色的过程类似，只是要将 MSP 路径的值替换为 order 节点上的 orderer.yaml 文件中的 'LocalMSPDir' 字段指定的值。

从其它 CA 服务器获取证书链

证书链，即是从直接执行认证行为的最下层 CA 机构，一直到最上层的根 CA 机构所经过的所有 CA 机构形成的线性依赖的证书集合。

通常 MSP 下存放证书的目录中，必须包含对当前节点来说可信任的、所有各级证书颁发机构的证书。

fabric-ca-client getcacerts 命令用于从其它 Fabric CA 服务端获取这些证书。

如下示例在本地启动了一个监听在 7055 端口、名为 "CA2" 的服务端，这展示了一个在区块链上由其它成员管理的完全独立的可信认证节点：

```
# export FABRIC_CA_SERVER_HOME=$HOME/ca2
# fabric-ca-server start -b admin:ca2pw -p 7055 -n CA2
```

如下命令将把 CA2 的证书安装到 peer1 的 MSP 路径下：

```
# export FABRIC_CA_CLIENT_HOME=$HOME/fabric-ca/clients/peer1
# fabric-ca-client getcacert -u http://localhost:7055 -M $FABRIC_CA_CLIENT_HOME/
msp
```

身份重新认证

如果证书将要过期，或者已经处于不安全状态，则需要重新认证获取新的证书：

```
# export FABRIC_CA_CLIENT_HOME=$HOME/fabric-ca/clients/peer1
# fabric-ca-client reenroll
```

证书与身份登记信息的撤销

身份登记信息和证书是可以被撤销的。撤销身份将使该身份所拥有的所有证书失效，并且会拒绝该身份申请新的证书；撤销证书只对单个证书有效。

执行撤销动作的角色必须具备 "hf.Revoker" 属性，并且其组织关系是与被撤销的对象处于同一部门或是其上级。例如：组织关系为 "orgs.org1" 的角色可以对组织关系为 "orgs.org1.department1" 的角色执行撤销动作，但不能对组织关系为 "orgs.org2" 的角色执行撤销动作。

如下命令撤销了一个身份（同时会撤销与其所拥有的所有证书），认身份未来的所有认证请求都会 Fabric CA 服务端拒绝：

```
# fabric-ca-client revoke -e <enrollment_id> -r <reason>
```

-r 选项可用的值有如下 10 项：

unspecified

keycompromise

cacompromise

affiliationchange

superseded

cessationofoperation

certificatehold

removefromcrl

privilegewithdrawn

aacompromise

如下，拥有最顶级组织关系的 admin 自认证身份，可以以如下方式注销掉 peer1 的身份：

```
# export FABRIC_CA_CLIENT_HOME=$HOME/fabric-ca/clients/admin
# fabric-ca-client revoke -e peer1
```

可以通过指定 AKI(Authority Key Identifier) 及其序列号来撤销单个证书：

```
fabric-ca-client revoke -a xxx -s yyy -r <reason>    #-s: serial number
```

某个证书的 AKI 与序列号可以通过 openssl 命令获取，如：

```
serial=$(openssl x509 -in userecert.pem -serial -noout | cut -d "=" -f 2)
AKI=$(openssl x509 -in userecert.pem -text | awk '/keyid/ {gsub(/ *keyid:|:/,"",
$1);print tolower($0)}')
```

```
fabric-ca-client revoke -s $serial -a $AKI -r affiliationchange
```

启用 TLS

以下将描述如何为 Fabric CA 客户端启用 TLS 支持。

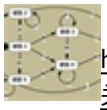
以类似如下的方式配置 fabric-ca-client-config.yaml 文件：

```
tls:
  # Enable TLS (default: false)
  enabled: true
  certfiles:
    - root.pem
  client:
    certfile: tls_client-cert.pem
    keyfile: tls_client-key.pem
```

certfiles 选项用于指定当前客户端信任的根证书集合，这通常就是 Fabric CA 服务端家目录下的 ca-cert.pem 文件。

client 部分的选项，只有当 Fabric CA 服务端启用了 TLS 双向认证策略时才需要。

HADEX_ FROM HELL.
分类: 区块链
标签: blockchain, fabric, hyperledger, 区块链



« 上一篇：Hyperledger：名词解释

» 下一篇：拾遗：git pull 与 push 远程分支与本地分支顺序识别问题