# 环境准备

## Fabric-ca环境准备

Fabric-ca的环境请参照《Fabric-ca集成》中的环境搭建，能使用默认配置启动fabric-ca-server则可以视为Fabric-ca部分的环境准备完毕。

## MySql环境准备

### Ubuntu下载安装mysql

安装Mysql

apt-get install mysql-server

apt-get install mysql-client

查看Mysql版本，若显示版本则说明安装成功

mysql -V

### 开放/修改3306端口（若无远程连接mysql跳过）

查看防火墙状态

sudo ufw status

如果防火墙状态时开启，可以选择关闭防火墙或者开启3306端口

关闭防火墙

sudo ufw disable

开启3306端口

sudo ufw allow 53

### 开放root远程登录权限（若无远程连接mysql跳过）

进入目录“/etc/mysql/mysql.conf.d/”，打开这个配置文件mysqld.cnf

需要加一条配置port = 3306，如果已经有，则不需要。

然后找到一条配置bind-address = 127.0.0.1，将此条配置注释掉。

取消了本地访问限制，我们后面对账户权限进行设置。

重启Mysql:

service mysql stop

service mysql start

输入以下命令，之后输入root密码进入mysql数据库:

mysql -u root -p

切换到系统数据库“mysql”中：

use mysql;

设置root用户任何IP都能登录数据库：

update user set host='%' where user='root';

开放root账户所有权限：

grant all privileges on \*.\* to 'root'@'%' identified by 'root账户密码';

使各种权限设置立即生效：

flush privileges;​

最后通过客户端测试连接数据库。

### 修改sql\_mode

由于Fabric-ca在使用mysql作为数据库，初始化建表时会出现timestamp默认值错误。所以需要修改sql\_mode。

输入以下命令，之后输入root密码进入mysql数据库:

mysql -u root -p

使用命令查看sql\_mode的值

select @@sql\_mode;

如果看到sql\_mode中有NO\_ZERO\_IN\_DATE和NO\_ZERO\_DATE则需要去除值。如修改root登录，进入目录“/etc/mysql/mysql.conf.d/”，打开这个配置文件mysqld.cnf

在此配置文件中添加一行，就是在默认sql\_mode默认值基础上去除以上两个值。

sql\_mode=ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_AUTO\_CREATE\_USER,NO\_ENGINE\_SUBSTITUTION

然后重启mysql，修改完毕。

# Fabric-ca服务器配置mysql数据库

### 获取和修改fabric-ca-server-config.yaml配置文件

在没有fabric-ca-server-config.yaml文件的时候，我们可以通过以默认配置启动Fabric-ca服务器来获得一个默认的fabric-ca-server-config.yaml文件。也可以通过新建yaml的方式得到fabric-ca-server-config.yaml配置文件。

新建一个文件夹：

mkdir -p /root/fabric-ca/server

使用启动默认配置CA服务器

fabric-ca-server init -b admin:adminpw

启动成功后会在当前目录下生成msp文件夹、db文件、fabric-ca-server-config.yaml文件，我们只需要保留fabric-ca-server-config.yaml文件，其余删除。

之后就是修改fabric-ca-server-config.yaml文件，改变CA服务器连接的数据库。找到文件中的“db:”选项，修改数据库连接配置，如下：

db:

type: mysql

datasource: root:123456@tcp(localhost:3306)/fabric?parseTime=true

affiliations项会让服务器在启动时检查数据库内组织，如果数据库无affiliations下的组织，则在启动时会添加以下组织。因为这里不需要默认的组织，所以将此项下的组织注释或者删除。

### 根据修改后的配置文件启动fabric-ca服务器

切换到/root/fabric-ca/server目录下，此目录下有上一步配置好的fabric-ca-server-config.yaml文件。然后我们用此配置文件启动CA服务器。因为需要执行添加或者删除组织和身份，所以在后面添加参数，如果不需要，则可以选择不添加。命令如下：

fabric-ca-server start -c fabric-ca-server-config.yaml -b admin:adminpw --cfg.affiliations.allowremove --cfg.identities.allowremove

然后看到输出的日志当中有如下，则说明根据数据库配置成功。

Initialized mysql database at \*\*\*\*:\*\*\*\*@tcp(localhost:3306)/fabric?parseTime=true

随后进入mysql中的fabric数据库，能看到创建的四张表：affiliations（组织）、certificates（证书）、users（用户）、properties（属性）。表里也有一些默认创建的信息。

### 向fabric-ca服务器添加组织和注册用户

执行以下命令添加组织：

fabric-ca-client affiliation add com

fabric-ca-client affiliation add com.example

fabric-ca-client affiliation add com.example.org1

fabric-ca-client affiliation add com.example.org2

执行以下命令创建用户:

fabric-ca-client register --id.name userexample --id.type user --id.affiliation com.example --id.attrs hf.Revoker=true

fabric-ca-client register --id.name userorg --id.type user --id.affiliation com.example.org1 --id.attrs hf.Revoker=true

### 查询mysql数据库储存的组织和用户

查询组织：

SELECT \* FROM affiliations;

查询用户：

SELECT \* FROM users;

查询结果得出对应表中新增组织和用户。

### 关闭后再根据fabric-ca-server-config.yaml配置文件启动检验

关闭CA服务器后，再次执行启动命令，然后再调用查询语句，得到数据库数据依然存在。然后再添加相同组织和用户。

fabric-ca-client affiliation add com

fabric-ca-client register --id.name userexample --id.type user --id.affiliation com.example --id.attrs hf.Revoker=true

添加失败，提示已经存在此组织和用户，说明CA服务器配置数据库成功。

# Fabric-ca集群