

FABRIC 入门

介绍:

本教程主要注重于 FABRIC 的环境搭建和第一个项目的实践，关于 Hyperledger Fabric 的基础理论知识，推荐大家可以去官方的中文社区学习：<https://hyperledgercn.github.io/hyperledgerDocs/> 由于中文社区的时效性，这里也推荐了英文的官方文档，有能力的读者可以去更好的去学习：https://hyperledger-fabric.readthedocs.io/en/release-1.4/key_concepts.html

FABRIC 依赖的环境：

DOCKER
DOCKER-COMPOSE
GO
PYTHON-PIP

环境搭建:

1. 前期准备:

个人服务器：（原先使用的 WMWARE 和 UBUNTU 使用，后有了服务器就放弃了，本教程也可以用虚拟机）

MobaXterm：用于将服务器可视化的工具，方便管理。

2. Go 的安装:

2.1 安装 GO 语言

```
root@ecs-819a-0009:~# wget https://dl.google.com/go/go1.9.2.linux-amd64.tar.gz
```

2.2 安装完成后，解压刚下载的文件

```
root@ecs-819a-0009:~# sudo tar -C /usr/local -xzf go1.9.2.linux-amd64.tar.gz
```

2.3 配置 GO 的环境变量

打开 profile 的文件：

```
root@ecs-819a-0009:~# vim ~/.profile
```

在文件末尾添加 4 个环境变量（C 插入）：

4 个 Export 的语句

```

if [ "$BASH" ]; then
    if [ -f ~/.bashrc ]; then
        . ~/.bashrc
    fi
fi
export PATH=$PATH:/usr/local/go/bin
export GOROOT=/usr/local/go
export GOPATH=$HOME/go
export PATH=$PATH:$HOME/go/bin
mesg n || true

```

2.4 编辑保存后（添加完成按 ESC 后 :wq 保存退出），使这些变量生效：

```

root@ecs-819a-0009:~# source ~/.profile

```

3. 安装 docker

FABRIC 一般运行的环境都是在 DOCKER 中，因此我们需要配置下 docker

3.1 检查 APT 是否有旧的 DOCKER，若有，则删除。

```

root@ecs-819a-0009:~# sudo apt-get remove docker docker-engine docker-ce docker.io

```

3.2 更新 apt 的索引包：

```

root@ecs-819a-0009:~# sudo apt-get update

```

3.3 安装插件使 apt 能通过 HTTPS 使用存储库（repository）：

```

root@ecs-819a-0009:~# sudo apt-get install -y apt-transport-https ca-certificates curl software-properties-common

```

3.4 添加 docker 官方的 GPG 密钥：

```

root@ecs-819a-0009:~# curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -

```

可以使用以下命令验证：

```

root@ecs-819a-0009:~# sudo apt-key fingerprint 0EBFCD88

```

3.5 设置 STABLE 存储库：

```

root@ecs-819a-0009:~# sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"

```

3.6 更新 apt 索引包：

```

root@ecs-819a-0009:~# sudo apt-get update

```

3.7 安装最新的 DOCKER-CE:

```
root@ecs-819a-0009:~# sudo apt-get install -y docker-ce
```

3.8 测试是否成功:

```
root@ecs-819a-0009:~# docker --version
Docker version 19.03.4, build 9013bf583a
root@ecs-819a-0009:~#
```

3.9 以下提供了一个快速安装 FABRIC 镜像的方法（使用阿里提供的镜像）:

在 docker 目录下新建 daemon.json 文件:

```
root@ecs-819a-0009:~# cd /etc/docker
root@ecs-819a-0009:/etc/docker# touch daemon.json
```

编辑新建的 daemon 文件输入以下内容:

```
"registry-mirrors":["https://obou6wyb.mirror.aliyuncs.com"]
```

编辑完成后，保存退出，并依次执行:

```
root@ecs-819a-0009:/etc/docker# sudo systemctl daemon-reload
```

```
root@ecs-819a-0009:/etc/docker# sudo systemctl restart docker
```

3.10 查看 docker 服务是否启动:

```
root@ecs-819a-0009:/etc/docker# systemctl status docker
```

4. docker-compose 的安装

4.1 安装 apt 的 docker-compose

```
root@ecs-819a-0009:/etc/docker# apt install docker-compose
```

4.2 验证是否安装成功

```
root@ecs-819a-0009:/etc/docker# sudo docker-compose --version
docker-compose version 1.8.0, build unknown
root@ecs-819a-0009:/etc/docker#
```

5. Fabric 源码下载:

5.1 先准备好存放和测试代码的目录:

```
root@ecs-819a-0009:/etc/docker# mkdir -p ~/go/src/github.com/hyperledger/
```

5.2 进入刚创立的目录中:

```
root@ecs-819a-0009:~# cd ~/go/src/github.com/hyperledger
root@ecs-819a-0009:~/go/src/github.com/hyperledger#
```

5.3 使用 git 命令在新建的目录中下载 FABRIC 源码:

```
root@ecs-819a-0009:~/go/src/github.com/hyperledger# git clone https://github.com/hyperledger/fabric.git
```

5.4 由于 FABRIC 源码的最新版一直不稳定, 本教程把源码切换到 v1.0.0 版:

```
root@ecs-819a-0009:~/go/src/github.com/hyperledger# git checkout v1.0.0
```

6. 下载 Fabric Docker 镜像:

以上我们完成了 Fabric 源码的下载, 下面将下载 Fabric 的 Docker 镜像。

6.1 cd 到 fabric 下的 examples 目录中, 并新建目录 e2e_cli

```
root@ecs-819a-0009:~/go/src/github.com/hyperledger# cd fabric
root@ecs-819a-0009:~/go/src/github.com/hyperledger/fabric# mkdir e2e_cli
```

6.2 下载 fabric 的 docker 镜像:

```
e2e_cli# source download-dockerimages.sh -c x86_64-1.0.0 -f x86_64-1.0.0
```

6.3 重新启动 docker:

```
es/e2e_cli# service docker restart
```

6.4 测试 fabric 环境是否成功:

```
hyperledger/fabric/examples# cd e2e_cli
hyperledger/fabric/examples/e2e_cli# ./network_setup.sh up
```

错误:

```
/opt/gopath/src/github.com/hyperledger/fabric/vendor/google.golang.org/grpc/cl
!!!!!!!!!! Channel creation failed !!!!!!!!!!!!!!!
===== ERROR !!! FAILED to execute End-2-End Scenario =====
```

目前还未找到解决方法, 正在询问其它组员和网上查找方法, 尝试解决。