数据区块链存储系统-部署文档

目录

[目录 1](#_Toc516331133)

[**一.部署环境** 1](#_Toc516331134)

[1.1服务器配置 1](#_Toc516331135)

[1.2客户机配置 1](#_Toc516331136)

[1.2.1客户端机器配置要求 1](#_Toc516331137)

[1.2.2客户端系统软件环境 1](#_Toc516331138)

[**二.系统安装与配置** 1](#_Toc516331139)

[2.1总体说明 1](#_Toc516331140)

[2.2启动网络 1](#_Toc516331141)

[2.3停止网络 1](#_Toc516331142)

[2.4重启网络 1](#_Toc516331143)

[2.5终止网络 2](#_Toc516331144)

[2.6实现区块链操作 2](#_Toc516331145)

[2.7调试网络 2](#_Toc516331146)

[**三.软件环境配置参考** 2](#_Toc516331147)

[3.1 cURL配置 2](#_Toc516331148)

[3.2 Docker配置 2](#_Toc516331149)

[3.3 Docker Compose配置 2](#_Toc516331150)

一.部署环境

1.1服务器配置

·服务器推荐使用Ubuntu 16.04 LTS作为服务器系统；

·服务器应保证至少有2G空余内存和10G左右的空闲硬盘空间；

·服务器应保证空闲时有至少1核左右的空余CPU；

·服务器应保证互联网通畅并拥有公网IP；

·服务器应保证有一块与公网IP相对应的网卡并且有空闲端口。

1.2客户机配置

1.2.1客户端机器配置要求

该项目运行环境对硬件设施要求不高，满足前端运行要求即可。

1.2.2客户端系统软件环境

·操作系统：MacOSX, Linux, Windows均可

·cURL：7.54.0及以上版本

·Docker：17.06.2-ce及以上版本

·Docker Compose：1.14.0及以上版本

·Go： 1.9.x及以上

二.系统安装与配置

2.1总体说明

启动Docker后，运行startFabric.sh脚本启动底层区块链网络，并使用BlockChainService接口实现对区块链的查询和调用。

2.2启动网络

启动Docker后，运行startFabric.sh脚本，该脚本将自动启动区块链网络，创建通道(Channel)，并将各个底层节点(Peer)添加到通道中。

2.3停止网络

运行脚本basic-network/stop.sh即可使网络停止。

2.4重启网络

当网络停止时，运行restart.sh脚本即可重启网络，区块链恢复正常工作状态。

2.5终止网络

若要在系统中彻底删除网络，运行脚本basic-network/teardown.sh即可。

2.6实现区块链操作

使用BlockChainService目录下的Java SDK可实现对区块链的操作，其中的接口提供了四个与区块链进行交互的方法，详细说明见《用户使用说明书》。

2.7调试网络

使用者可以使用docker exec命令进入CLI容器中进行网络的调试。

三.软件环境配置参考

3.1 cURL配置

cURL一个利用URL规则在命令行下工作的文件传输工具，支持文件的上传和下载。

可在https://curl.haxx.se/download.html获取对应操作系统的发行版本，安装后即可使用。

若要使用Windows操作系统，可以给Windows增加curl命令的环境变量，增加CURL\_HOME环境变量，给PATH环境变量加上%CURL\_HOME%;。即可在命令窗口的任意目录下使用curl命令。

3.2 Docker配置

Docker 是一个开源的应用容器引擎，让开发者可以打包他们的应用以及依赖包到一个可移植的容器中，以简化部署多种应用实例工作。

以Ubuntu为例：

获取最新版本的 Docker 安装包：$ wget -qO- https://get.docker.com/ | sh

启动docker 后台服务：$ sudo service docker start

3.3 Docker Compose配置

Docker Compose是运行多容器Docker应用程序的工具。

以Ubuntu为例：pip安装：$ sudo pip install docker-compose

安装curl后也可：

# curl -L https://github.com/docker/compose/releases/download/1.14.0/docker-compose-`uname -s`-`uname -m` > /usr/local/bin/docker-compose