**基于区块链技术的供应链平台**

**区块链数据存储**

**部署文档**

**北航软件工程课程实践**

**二○一九年六月**

**北航软件工程课程实践**

**二○一九年三月**

目录

[一.部署环境 1](#_Toc11423461)

[1.1服务器配置 1](#_Toc11423462)

[1.2客户机配置 1](#_Toc11423463)

[1.2.1客户端机器配置要求 1](#_Toc11423464)

[1.2.2客户端系统软件环境 1](#_Toc11423465)

[二.系统安装与配置 1](#_Toc11423466)

[2.1总体说明 1](#_Toc11423467)

[2.2启动网络 1](#_Toc11423468)

[2.3停止网络 2](#_Toc11423469)

[2.4重启网络 2](#_Toc11423470)

[2.5终止（删除）网络 2](#_Toc11423471)

[2.6实现区块链操作 2](#_Toc11423472)

[2.7调试网络 2](#_Toc11423473)

[2.8 调用java应用以启动区块链服务 2](#_Toc11423474)

[三.软件环境配置参考 2](#_Toc11423475)

[3.1 cURL配置 2](#_Toc11423476)

[3.2 GO安装配置 3](#_Toc11423477)

[3.3 Docker配置 3](#_Toc11423478)

[3.4 Docker Compose配置 3](#_Toc11423479)

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

硬件设施需要能够满足前端运行要求。

1.2.2客户端系统软件环境

·操作系统：Mac OSX, \*nix, Windows(需在WSL内)

·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: 1.12.x及以上

·JRE 8.0 及以上

二.系统安装与配置

2.1总体说明

在管理员权限下，启动Docker后，运行basic-network/start.sh脚本启动底层区块链网络，并使用BlockChain接口对区块链进行查询与调用。

2.2启动网络

启动Docker后，运行basic-network/start.sh脚本，运行basic-network/start.sh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

若要使得网络停止，则运行脚本basic-network/stop.sh。

2.4重启网络

若要在网络停止时重启网络，则运行basic-network/restart.sh脚本即可，重启网络后区块链恢复正常工作状态。

2.5终止（删除）网络

运行脚本basic-network/teardown.sh即可在系统中彻底终止（删除）网络。

2.6实现区块链操作

使用BlockChain目录下的Java SDK可实现对区块链的操作，其中的接口提供了十个与区块链进行交互的方法，详见《用户使用说明书》。

2.7调试网络

通过docker exec -it cli bash命令进入CLI容器调试网络。

2.8 调用java应用以启动区块链服务

对编译好的jar文件夹Blockchain\_jar.zip 进行解压；

进入文件目录后，执行命令 java –jar Blockchain.jar即可调用测试代码。

三.软件环境配置参考

3.1 cURL配置

cURL是一个基于URL规则在命令行下工作的文件传输工具，支持上传和下载文件。

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 GO安装配置

Go（又称Golang）是Google开发的一种静态强类型、编译型、并发型，并具有垃圾回收功能的编程语言。

下载对应系统的安装包并将其解压缩到 /usr/local，在 /usr/local/go中创建一个Go树。 例如：

tar -C /usr/local -xzf go$VERSION.$OS-$ARCH.tar.gz

注意，如果要在Linux上为64位x86安装Go版本1.2.1，则所需的存档称为go1.2.1.linux-amd64.tar.gz，以此类推（通常这些命令必须以root身份或通过sudo运行。）

将 /usr/local/go/bin添加到PATH环境变量。 此操作可以通过将下面这行指令添加到/etc/profile（用于系统范围的安装）或 $HOME/.profile来实现：

export PATH=$PATH:/usr/local/go/bin

3.3 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，然后发布到任何流行的 Linux 机器上，而且也可以实现虚拟化。通过Docker可以简化部署多种应用实例工作。

对于Docker的配置，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

3.4 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后也可使用以下命令安装配置：

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