**Fabric编译制作镜像及测试**

**fabric-baseimage**

下载fabric-baseimage项目

git clone <https://github.com/hyperledger/fabric-baseimage.git>

git checkout v0.4.10

将fabric-baseimage/scripts/common/setup.sh中下载golang的地方改为下载ftp中的国密版golang

# Install Golang binary if found in BINTARGETS

if echo $BINTARGETS | grep -q `uname -m`; then

cd /tmp

wget ftp://192.168.20.211/golang/go-gm1.10.3.linux-amd64.tar.gz

tar -xvf go-gm1.10.3.linux-amd64.tar.gz

mv go $GOROOT

chmod 775 $GOROOT

# Otherwise, build Golang from source

else

在fabric-baseimage目录下make docker制作基础镜像

**fabric**

Go Tools安装

mkdir -p $GOPATH/src/golang.org/x

cd $GOPATH/src/golang.org/x

git clone <https://github.com/golang/tools.git>

git clone <https://github.com/golang/lint.git>

git clone [https://github.com/golang/crypto.git](https://github.com/golang/tools.git)

git clone https://github.com/golang/sys.git

下载完毕，安装Fabric可能用到的Go工具：

go get github.com/kardianos/govendor

go get github.com/golang/lint/golint

go get golang.org/x/tools/cmd/goimports

go get github.com/onsi/ginkgo/ginkgo

go get github.com/axw/gocov/...

go get github.com/client9/misspell/cmd/misspell

go get github.com/AlekSi/gocov-xml

go get github.com/golang/protobuf/protoc-gen-go

go get github.com/estesp/manifest-tool

go get github.com/maxbrunsfeld/counterfeiter

go get github.com/vektra/mockery/cmd/mockery

分步编译（用fabric-samples测试时需要用到分步编译生成的几个可执行文件，所以这里采用分步编译）

cd $GOPATH/src/github.com/hyperledger/fabric

mkdir -p .build/docker/gotools/bin

cp $GOPATH/bin/\* .build/docker/gotools/bin （此前安装的gotools可能会由于go环境变量配置的原因而存在$GOPATH/bin或者$GOROOT/bin里面，此处路径对应填写）

把Makefile文件中下载chaintool的路径改掉，或者直接用原来的路径，只是原路径从网上下载有可能会失败

CHAINTOOL\_URL ?= ftp://192.168.20.211/fabric\_compile\_deps/hyperledger-fabric-chaintool-1.1.1.jar

1.编译orderer

make orderer

2. 编译peer

make peer

3. 编译tools

make configtxgen

make cryptogen

make configtxlator

4.生成镜像

make orderer-docker

make peer-docker

make tools-docker

make docker

直接编译可如下操作

cd $GOPATH/src/github.com/hyperledger/fabric

mkdir -p .build/docker/gotools/bin

cp $GOPATH/bin/\* .build/docker/gotools/bin

make docker

**fabric-samples**

下载fabric-samples项目

git clone <https://github.com/hyperledger/fabric.git>

在fabric-samples中创建文件夹bin

cd $GOPATH/src/github.com/hyperledger/fabric-samples

mkdir bin

把fabric/.build/bin/里面的二进制文件复制到fabric-samples/bin/里面

cd $GOPATH/src/github.com/hyperledger/fabric/.build/bin

cp \* $GOPATH/src/github.com/hyperledger/fabric-samples/bin/

进入first-network里面在crypto-config.yaml文件中最后面添加一个配置，根据配置决定CA以及签发的证书是SM2还是ECDSA的：

HashFamily: SM3

执行脚本byfn.sh

./byfn.sh -m up