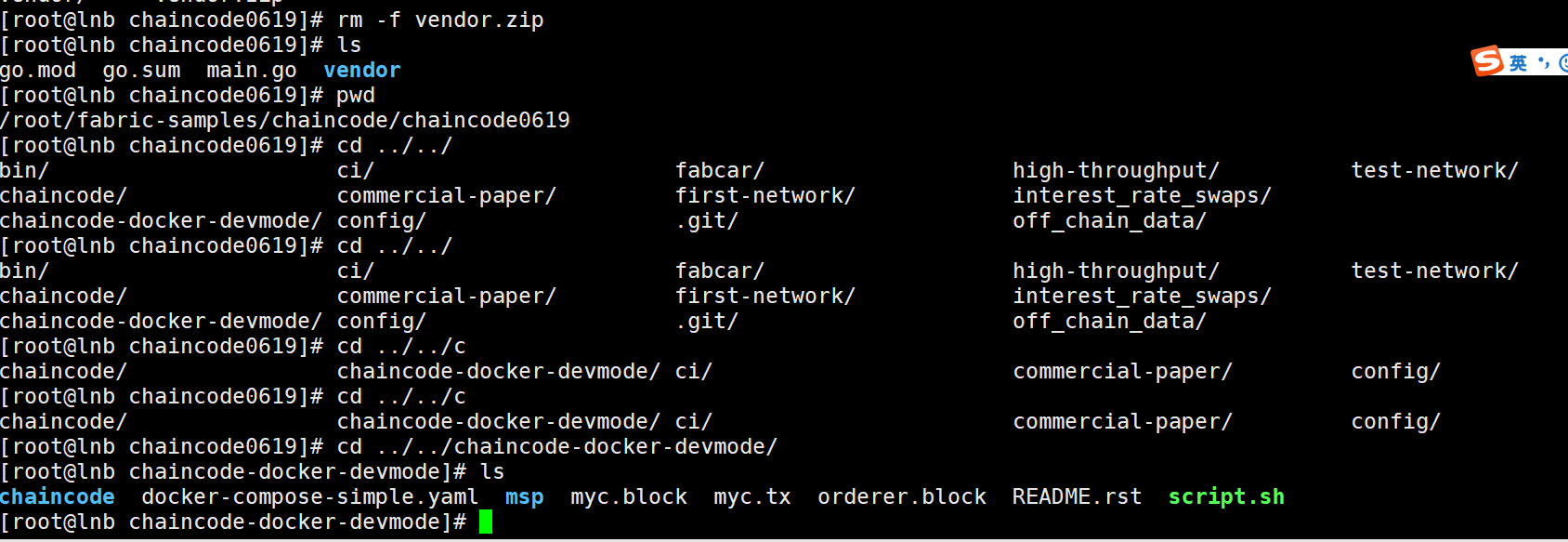
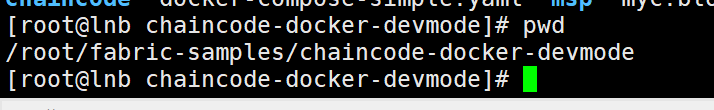
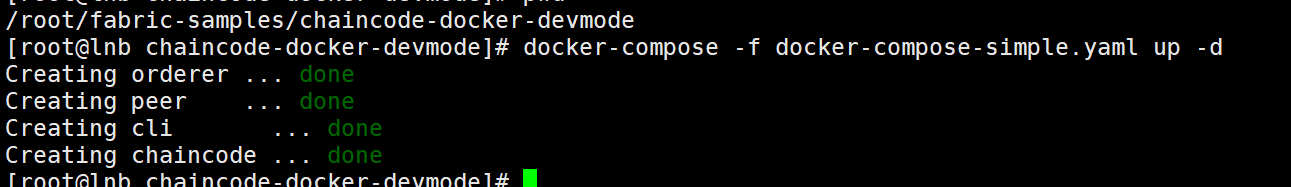
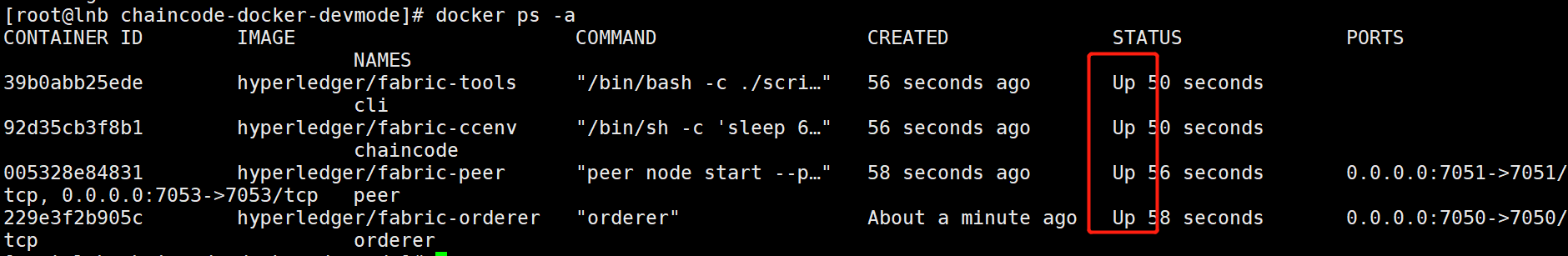
1. 启动虚拟机
2. 启动Xshell，链接虚拟机（Fa）
3. ping baidu.com 有回应Ctrl +c，没回应重启忘了systemctl restart network
4. 启动至少三个窗口，切换窗口 alt + 窗口号
5. cd ~/fabric-samples/chaincode（Cd 切换目录 ls查看当前目录所含文件）
6. 创建文件夹mkdir + 目录名 并且切换目录
7. 拖拽代码文件（四个）
8. Unzip + 文件名 解压文件（.zip）
9. Rm -f + 文件名 （.zip文件）(rm -fr + 文件名 （删除目录）)
10. 
11. 切换到启动容器所在目录(不切换窗口)



1. 启动docker容器 切换目录并启动



1. 查看是否启动成功



如果启动不成功：

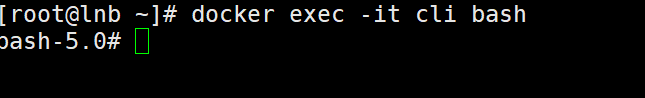
重启未成功的



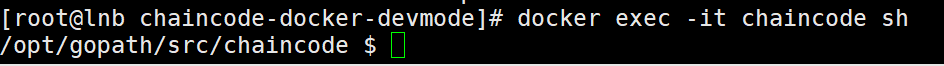
强制关闭所有



1. 进入docker容器（切换shell窗口）



切换窗口，进入chaincode容器

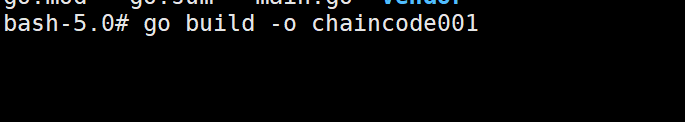


14.切换到cli窗口，切换目录

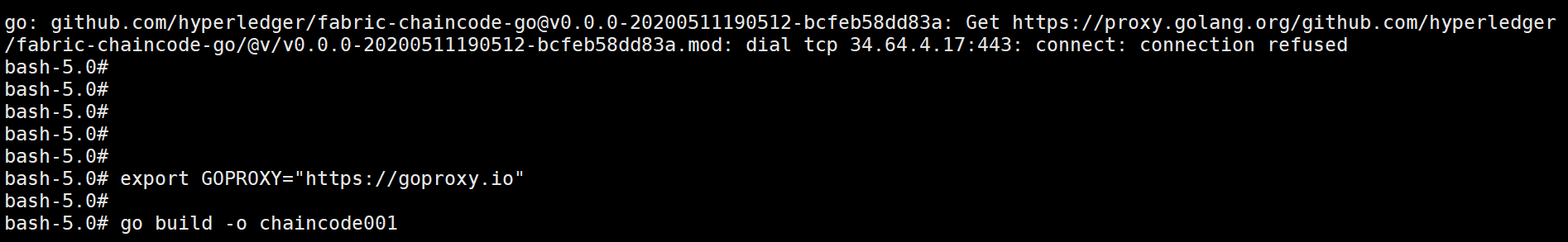


15.切换代理 export GOPROXY=<https://goproxy.io>

16.构建

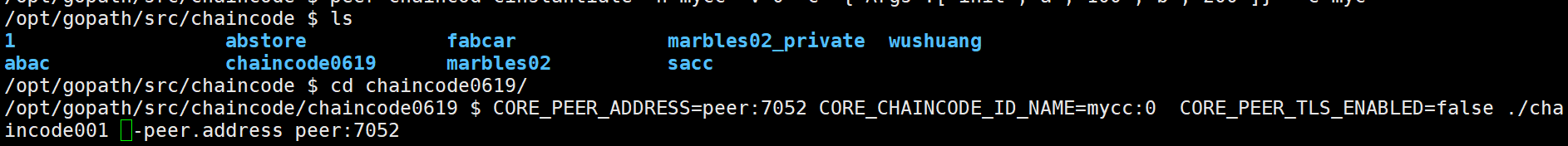


17.报错之后改代理，重新构建ls



18.切换chaincode窗口 启动链码（切换到指定目录）

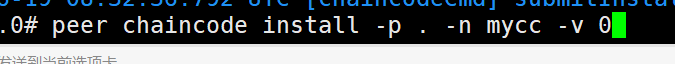
CORE\_PEER\_ADDRESS=peer:7052 CORE\_CHAINCODE\_ID\_NAME=wushuang:0 CORE\_PEER\_TLS\_ENABLED=false ./chaincode001 -peer.address peer:7052



19. 切换cli窗口，安装链码peer chaincode install -p 目录名 -n mycc -v 0

peer chaincode install -p . -n mycc -v 0

peer chaincode install -p . -n wushuang -v 0



注意：当前所在目录 pwd查看

20.实例化链码

peer chaincode instantiate -n mycc -v 0 -c '{"Args":["createFileChaincode","1","asdfasdfasdfasdfg","模型","asdfgh","zxcvb"]}' -C myc

peer chaincode instantiate -n wushuang -v 0 -c '{"Args":["createFileChaincode","1","asdfasdfasdfasdfg","模型","asd","zxc"]}' -C myc



21.查询等操作

peer chaincode query -n mycc -c '{"Args":["readFileChaincode","1"]}' -C myc



添加peer chaincode invoke -n mycc -c '{"Args":["createFileChaincode","1","afsdfasdfasdfasdfg","模型","asd","zxc"]}' -C myc



queryStr := fmt.Sprintf("{\"selector\": {\"ToUser\":\"%s\"}}", toUser)

queryStr := fmt.Sprintf("{\"selector\": {\"ObjectType\":\"fileChaincode\",\"toUser\":\"%s\"}}", toUser)

peer chaincode instantiate -n wushuang -v 0 -c '{"Args":["createFileChaincode","1","asdfasdfasdfasdfg","模型","asd","zxc"]}' -C myc

peer chaincode invoke -n wushuang -c '{"Args":["createFileChaincode","1","afsdfasdfasdfasdfg","模型","asd","zxc"]}' -C myc

peer chaincode invoke -n wushuang -c '{"Args":["deleteFileChaincode","1","afsdfasdfasdfasdfg","模型","asd","zxc"]}' -C myc

peer chaincode query -n wushuang -c '{"Args":["getHistoryFromFileChaincode","1"]}' -C myc

peer chaincode query -n wushuang -c '{"Args":["queryFileChaincodeByOwner","zxc"]}' -C myc

peer chaincode query -n wushuang -c '{"Args":["readFileChaincode","1"]}' -C myc

peer chaincode query -n wushuang -c '{"Args":["queryFileChaincodeByTxID","38217bbc2e781c51feac7bf3f7b7350708ed1245b918af89fbcd143e482c006e"]}' -C myc

peer chaincode invoke -n wushuang -c '{"Args":["transferFileChaincode","3","afsdfasdfasdfasdfg3","zxc3","asd3"]}' -C myc

89ad6de727fd2e334687df6631bd0b4bea227217a0b43d07a590590f1619106f