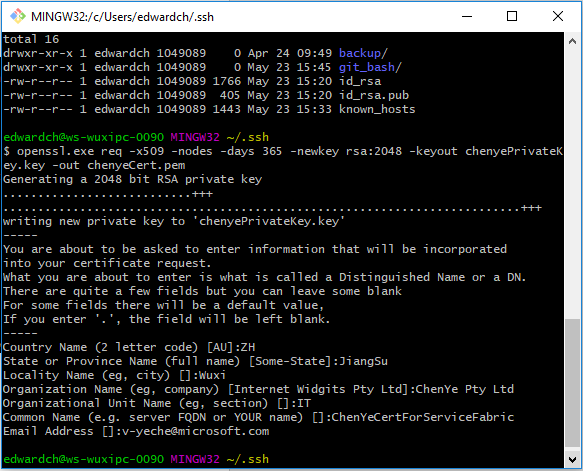
Install git and run the terminal git bash

openssl.exe req -x509 -nodes -days 365 -newkey rsa:2048 -keyout edwardPrivateKey.key -out edwardCert.pem

openssl pkcs12 -in party-cluster-1486790479-client-cert.pfx -out party-cluster-1486790479-client-cert.pem -nodes -passin pass:1486790479

-nodes 表示私钥不加密

openssl.exe req -x509 -days 365 -newkey rsa:2048 -keyout edwardPrivateKey.key -out edwardCert.pem



EdwardCommonNameServiceFabric

chmod 0600 edwardPrivateKey.key

openssl.exe rsa -pubout -in edwardPrivateKey.key -out edwardPublicKey.key

openssl.exe x509 -outform der -in edwardCert.pem -out edwardCert.cer

openssl x509 -fingerprint -in edwardCert.cer

基本语法：

openssl x509 -fingerprint -in cerfile.crt

若是报无法读取文件，那么极有可能是未指定格式，可尝试如下

openssl x509 -fingerprint -in cerfile.crt -inform PEM

openssl x509 -fingerprint -in cerfile.crt -inform DER

若是需要指定显示SHA1或者SHA256算法编码的指纹信息，则追加参数

openssl x509 -fingerprint -sha1 -in cerfile.crt

openssl x509 -fingerprint -sha256 -in cerfile.crt



SHA1 Fingerprint=4B:DB:67:AE:7C:E4:01:7B:1B:84:4C:FC:D6:18:B2:12:7B:22:29:8A

Reference 2

1、生成.key文件

openssl genrsa -des3 -out server.key 2048

**openssl genrsa -des3 -passout pass:Lindsayc5 -out server.key 2048**

中间会提示输入密码(重复输入两次)，要记住这个密码；

这时会在C:\OpenSSL-Win64\bin目录下生成server.key文件。

2、生成.crt文件

openssl req -new -x509 -key server.key -out server.crt -days 365

//with password type in

**openssl req -new -x509 -passin pass:Lindsayc5 -key server.key -out server.crt -days 365**

会提示输入server.key的密码

开始输入Country Name：CN

State or Province Name：SH

Locality Name：shanghai

Organization Name：这个可以忽略

Organizational Unit Name：这个可以忽略

Common Name：chenyecluster.chinanorth.cloudapp.chinacloudapi.cn

Email Address：填写一个非QQ的邮箱地址

这时会在C:\OpenSSL-Win64\bin目录下生成server.crt文件。

3、生成.pfx文件

openssl pkcs12 -export -out server.pfx -inkey server.key -in server.crt

**openssl pkcs12 -export -out server.pfx -password pass:Lindsayc5 -passin pass:Lindsayc5 -inkey server.key -in server.crt**

提示输入server.key文件的密码

提示输入即将生成的.pfx文件的密码(需要输入两次)

这时会在C:\OpenSSL-Win64\bin目录下生成server.pfx文件。

---------------------

作者：hu\_gang

来源：CSDN

原文：https://blog.csdn.net/hu\_344/article/details/73368774

版权声明：本文为博主原创文章，转载请附上博文链接！

openssl pkcs12 -in party-cluster-1486790479-client-cert.pfx -out party-cluster-1486790479-client-cert.pem -nodes -passin pass:1486790479

$ResourceGroupName="ChenYeRG"

$ClusterName="chenyecluster"

$Location="chinanorth"

$Password="RyanLindsayc5"

$VaultName="chenyevault"

$VaultGroupName="ChenYeKVaultRG"

$CertPath="C:\mycertificates"

az sf cluster create --resource-group $ResourceGroupName --location $Location `

--cluster-name $ClusterName --certificate-subject-name chenyecluster.chinanorth.cloudapp.chinacloudapi.cn --template-file vnet-linuxcluster.json --parameter-file vnet-linuxcluster.parameters.json `

--certificate-password $Password --certificate-output-folder $CertPath `

--vault-name $VaultName --vault-resource-group $ResourceGroupName

IMPORTANT:

1. Generate the self sign certificate.
   1. **openssl genrsa -des3 -passout pass:Lindsayc5 -out server.key 2048**
   2. **openssl req -new -x509 -passin pass:Lindsayc5 -key server.key -out server.crt -days 365**
   3. **openssl pkcs12 -export -out server.pfx -password pass:Lindsayc5 -passin pass:Lindsayc5 -inkey server.key -in server.crt**
2. **Deploy the service fabric with template and parameter file.**

$ResourceGroupName="ChenYeRG"

$Location="chinanorth"

$Password="Lindsayc5"

$VaultName="chenyevault"

$VaultGroupName="ChenYeKVaultRG"

$CertPath="C:\mycertificates\server.pfx"

# sign in to your Azure account and select your subscription

az login

az account set --subscription <guid>

# Create a new resource group for your deployment and give it a name and a location.

az group create --name $ResourceGroupName --location $Location

# Create the Service Fabric cluster.

az sf cluster create --resource-group $ResourceGroupName --location $Location `

--certificate-password $Password --certificate-file $CertPath `

--vault-name $VaultName --vault-resource-group $VaultGroupName `

--template-file vnet-linuxcluster.json --parameter-file vnet-linuxcluster.parameters.json