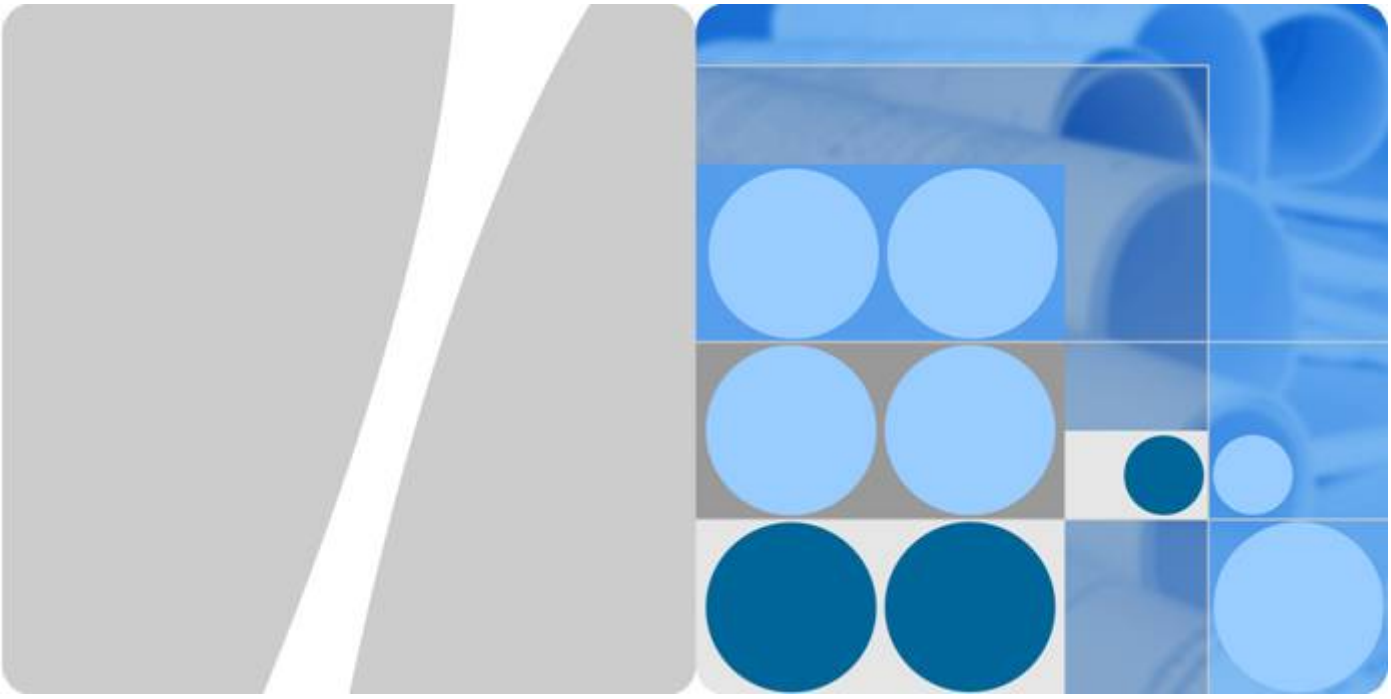


Ascend 310

V100R001

Mind Studio 工具证书替换指导

文档版本　01

发布日期　2019-03-12

华为技术有限公司

HUAWEI

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： http://www.huawei.com

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

目 录

1 简介

密钥证书使用场景

- 用于Mind Studio（作为服务端）和Google浏览器（作为客户端）之间的单向认证。
- 用于IDE-daemon-host（作为服务端）和Mind Studio（作为客户端）之间的双向认证。

密钥证书列表

Mind Studio安装完成后，会默认自带表1-1中的证书或密钥，您可以以Mind Studio安装用户登录Mind Studio服务器，在“~/tools/conf/secure_keys”目录下查看表1-1中的证书或密钥。其中，Huawei_2012_laboratory_CA.cer和Huawei_Equipment_Root_CA.der需要浏览器侧导入，先导入Huawei_Equipment_Root_CA.der，再导入Huawei_2012_laboratory_CA.cer。

说明

“~/tools”是默认的toolpath路径，该路径可在安装Mind Studio时由用户自定义，您可以在“scripts/env.conf”文件通过toolpath参数查看实际路径。您可以使用**find / -name 'env.conf'**命令查看script目录下的“env.conf”文件的位置。

表 1-1 证书/密钥列表

存放位置	证书或密钥名称	用途	是否加密
服务端	server.key	server端私钥	是
服务端	server.crt	server端证书	否
服务端	server.p12	由server端证书和私钥生成的PKCS12格式的密钥库中	是

存放位置	证书或密钥名称	用途	是否加密
客户端（浏览器）	CA/Huawei_2012_laboratory_CA.cer	华为CA二级根证书	否
客户端（浏览器）	CA/Huawei_Equipment_Root_CA.der	华为CA根证书	否
客户端	client.key	client端私钥	是
客户端	client.crt	client端证书	否
客户端	client.p12	client端证书和私钥生成的PKCS12格式秘钥库	是
客户端/服务端	catrust.jks	CA信任证书库	是
客户端/服务端	ca.crt	CA证书	否

证书获取方式

您可以根据实际需求替换默认自带的证书或密钥，在替换证书前，您需要先获取证书，获取证书有以下几种方式：

- 测试时，使用操作系统自带的openssl工具和JDK自带的keytool工具自制证书和密钥，包括服务端和客户端的证书、密钥。
- 商用时，向第三方机构申请安全证书。

说明

如果是自制证书，则不需要同时制作Huawei_2012_laboratory_CA.cer和Huawei_Equipment_Root_CA.der两个文件，只需制作一个CA.cer文件，导入到浏览器中即可。

2 获取证书

2.1 自制证书和密钥（测试）

2.1.1 自制服务端的证书和密钥

您可以使用JDK自带的keytool工具自制服务端的密钥库文件（server.p12），再使用openssl工具将server.p12转换为server.pem，最后根据server.pem文件得到服务端的私钥文件（server.key）、证书文件（server.crt）。

步骤1 以Mind Studio的安装用户登录Mind Studio服务器。

步骤2 生成服务端的keystore文件“server.p12”。

在当前目录或切换到其它目录下，执行以下命令生成keystore文件。

1. 执行keytool命令。
```
keytool -genkey -storetype PKCS12 -keystore server.p12 -alias 1 -keysize 2048 -keyalg RSA -ext san=ip:xx.xx.xx.xx -validity 3650
```
在以上命令中，各参数的解释如下：
   - **-genkey**表示生成一个钥匙对（公钥和私钥）。
   - **-storetype PKCS12**表示指定密钥库的类型为PKCS12。
   - **-keystore**参数后面跟着的是 keystore文件的名字，可修改。
   - **-alias**参数后面跟着的是生成的密钥和证书的别名，可修改。
   - **-keysize**参数后面跟着算法长度。
   - **-keyalg**参数后面跟着生成密钥对的算法。
   - **-ext san**参数用于绑定服务器的IP地址。ip需要设置为Mind Studio服务器的IP。*xx.xx.xx.xx*需要替换为实际Mind Studio服务器的IP。
   - **-validity**参数后面跟着证书的有效期，以天为单位。如果不指定有效期，则默认有效期是90天。
2. 输入keytool命令后，回显信息如下，您需要根据提示信息输入，输入的字符串长度大于等于6且小于等于15。
```
Enter keystore password:（输入keystore的访问密码）
Re-enter new password: （重复输入keystore的访问密码）
```
3. 输入密码后，您需要根据表2-1中的回显信息的提示，输入相应的信息。

表 2-1 回显信息和输入信息

顺序	回显信息	输入信息
1	What is your first and last name?	这里必须是Mind Studio服务器IP地址。
2	What is the name of your organizational unit?	根据待启用证书服务器实际所在的公司部门名称输入，例如“Huawei HISI”。
3	What is the name of your organization?	根据待启用证书服务器实际所在的公司名称输入，例如“Huawei”。
4	What is the name of your City or Locality?	根据待启用证书服务器实际所在的城市名输入，例如“HangZhou”。
5	What is the name of your State or Province?	根据待启用证书服务器实际所在的国家名称输入，例如“China”。
6	What is the two-letter country code for this unit?	根据待启用证书服务器实际所在的2位国家缩写代码输入，例如“CN”。

4. 输入完成后，系统将询问用户输入的信息是否正确，如果正确，请输入“y”或“yes”，否则请输入“n”或“no”。
5. 输入“y”或“yes”后，系统提示输入访问密钥的密码，根据提示输入密码。
6. 执行以上命令后，如果系统不提示错误，则可以在当前目录下查看生成的keystore文件“server.p12”。

步骤3 生成服务端的加密私钥server.key和证书server.crt。

1. 将“server.p12”文件转化成“server.pem”文件。

```
openssl pkcs12 -in server.p12 -out server.pem
```

在以上命令中，各参数的解释如下：

- **-in**：该参数后面跟着的是pkcs12文件的路径，包含文件名。
- **-out**：该参数后面跟着的是证书或私钥文件的路径，包含文件名，文件名格式必须是*.pem。

openssl命令执行后，系统会回显如下信息，您需要根据提示输入密码，输入密码后，在执行openssl命令的目录下会生成server.pem文件。

```
Enter Import Password:
MAC verified OK
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
```

在“Enter Import Password”处输入访问密钥库文件**server.p12**的密码，与**步骤2.2**中输入的密码一致。

在“Enter PEM pass phrase”和“Enter PEM pass phrase”处输入访问**server.pem**的密码，与**步骤2.2**中输入的密码一致。

2. 创建一个空文件，命名为server.key。

打开server.pem，复制“-----BEGIN ENCRYPTED PRIVATE KEY-----”到“-----END ENCRYPTED PRIVATE KEY-----”之间字符串（加密后的密钥），保存到server.key文件中。


```
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFDjBABgkqhkiG9w0BBQ0wMzAbBgkqhkiG9w0BBQwwDgQI4AEL+4wQyJkCAggA
MBQGCCqGSIb3DQMHBAgTjDnqymFRfQSCBMhDfokWIC9TiSpEBlIh4XIQW1z9oJrw
hlP767ZgUFh/8FRimAOmpFbeN0tUwdbd4kHJJ34soRdEQMb8HqWdIK5Ys9NnxYyR
mepddHSc/WMxegchy4+MDVs5ksFuabXohKd8rDDHHndRgPGSsS5s1+DEjLQiww5E
iJ+ACUhWZxMbJ+idiNEl/oaB0dh2pL/AdVIVmbRJnvTwSkpVUznmV33+1fGyeSht
z6uHlQCkFk+F/DqerhfUtENgV3FeM1rJTYbaKW14DGOg1JsD8ARtQ5PQd2N+30Te
6Ax81/OgDJmALa9stwyJjQHZvFPzmmk+ju4vkcGCMLSxgB8Qa8g569NY1CadpveD
fADl0AVHqeuvWtsZDGVXlew/4HeSUXxOxebSb5LZ3MCBKvJCACKXpESQvd1ajkvF
yEw2T7YHDa+GIHeuu//F2rWHFGGpH25GZXC1PlPfPiIZONSf5jDgEPZb9du/k9Oc
zKg+xKp2E2KBN03bJRvJjVy8M8vhfinrtqN4I8bPeuqJFMkdaRFVIkWfg6WEfrDR
gwjv4RGV69H5kJZ84EUD7JycXaL+AsU2h/r5db1NVO6ZmQLb5ZcW+hPdLWIOWZ6i
agyY29YNvhPpcvoRkn+F1AAkMsWcW8G5Ck3BQoNtRl1PMM7zvgGAvbhWGPlQPX0o
hQnk6hyGRr8wdmX8El9qp8nRrUYGeqcstobYVqlrRhZPC3ItCKvG/90JLxtkMT7x
hjTWg6mcxawBMWFyY/o8LoblLVSDez9IwICUsKR6K4RKob3Xsi9BP4kQd1qvAVNU
BdOeG0HYhIoPWSbckzVHQd6p2Wm50Lrc/9P5UseyxfeMUD10rmlH4UPSMTztlYyU
FxkM01LFEBNaVT26fmBa1CseDIgOHwgI9hWOu6meyRoSExAF2C8I7TFAe0jql2fs
q+RZ9/4Mm717XTVzsTVZg3BIsQCECdclIDMmhSFwXjTveuOrlsrbLfX6/8Cma0PV
OPAkeVFCgGMkbZHGQxOTb7OTXvHNaImaOlTWX1NbxI23YTqa/iTptbJqXHGGD6DR
TtTM1SQKDCl2quFK74FXyRvben9tUzxFHulEGDt1vQhkl9w3c+z4QvCQ7OO4FLGD
bEaf4keA1GuK5p16jwHNf7DeQjK5x+n45HvuDdg9L7t7/UbTXDsanfFJLV7oDFJb
AYLvbsSbfSI8xtg3PC8q2Ir/ZLaIatFY799t/uTqrBEpv9MScBhNq9LmX4IOFnhM
KY4CgXN/xM7zv4nB+et5Vcl2VgQ4kM/UoeQO3D9ynGi85Ssz0SYEgHlqoha/xwrs
x2JCGcfy94b7XNV4jWNmnn1skSfJbfVKjLF3I++SIUSWAiSnBOBJt6qd3CzCUDgf
IVy8NS/F2vq4hwzhYeyeSuBNXOwtHxSZ8mCiUA3uTGESswe3E+ORrcnke7FdL0Kk
GOKSexsaGP+vClPfIIIiBz/ZQkjMr6MHN0Mui/mbFa5qGm/6BOsOHbmtftRcTZhG
vShQcl5EQ5aoEqqQwhtZTJ0qFk+tiDqHaeKnKuDyzzUqjWXFCFboujKV4uK0c4D/
Cpj7VcYIWqcGc3H2+fCUkP7/a+FUXQg6ZHvfPF0APvdziaW05Xu7Oapgrp/Gmf0T
FBk=
-----END ENCRYPTED PRIVATE KEY-----



server.key" 30L, 1834C
```

3. 创建一个空文件，命名为server.crt。

   打开server.pem，复制“-----BEGIN CERTIFICATE-----”到“-----END CERTIFICATE-----”之间字符串（证书），保存到server.crt文件中。


```
-----BEGIN CERTIFICATE-----
MIIDfDCCAmSgAwIBAgIEP+0LZDANBgkqhkiG9w0BAQsFADBmMQswCQYDVQQGEwJD
TjELMAkGA1UECBMCemoxDzANBgNVBAcTBmh1YXdlaTEPMA0GA1UEChMGaHVhd2Vp
MQ8wDQYDVQQLEwZodWF3ZWkxFzAVBgNVBAMTDjEwLjE3OS4xNzEuMTE1MB4XDTE5
MDExNjAzMjI0NVoXDTE5MDQxNjAzMjI0NVowZjELMAkGA1UEBhMCQ04xCzAJBgNV
BAgTAnpqMQ8wDQYDVQQHEwZodWF3ZWkxDzANBgNVBAoTBmh1YXdlaTEPMA0GA1UE
CxMGaHVhd2VpMRcwFQYDVQQDEw4xMC4xNzkuMTcxLjExNTCCASIwDQYJKoZIhvcN
AQEBBQADggEPADCCAQoCggEBAIg/ifQ1lXiA6C3KkHFqRrjRp9dRpSz7pABeTYa6
+NDnACt7yhzsV42hG5l/X39NmzJYksBi3b4gqc3UeDVYHDt/JPwRzB5CXLXSjYJa
sKJX0eJ1a9I9GXK6nlC6fFfrysl3yENoNgl/recSZ5+6x8GoOwm4j5AehrzBhNkJ
oGFApKsRRih1St5kxnKjfN2ylHP+2YRMs+8IG+3aAy501murU2qITj733Qfh9wb5
FmA+7EgGSQnMNHWtIU20jvoAilIrWlwHjODY0k3HVaNkEMjrZgQVcDTL/7dz5DLA
eq5Mv/SwFS8Pa9GzY1bBHo25ob8ZVwqi7BhjYLSAaYLKQUkCAwEAAaMyMDAwDwYD
VR0RBAgwBocECr0rczAdBgNVHQ4EFgQU6BDlrWQDAJWp7dJSy/xuAOKRhLwwDQYJ
KoZIhvcNAQELBQADggEBAGoXpZWQn1dSgUzIH6CLojwUGbs3LqQiwvb54vokfOsb
+BzQIX1Fd7UKyfk3V+lVD6txw1wGwMANk7yLDp+85qKRWEysCkoV+7kjr2QqpmmP
xf6QVeA5GzlXLbFqgRTLbsiHJLiTJ6K1PngTOFqJmyDNORDGw+pIxSzz4OfzG148
cPzH05zw65xpj/LRTqJGEyTyDIBKVRZlOD4egv3gdqU17Z/LRopf7MHCaTHddZFL
FQ1gSRK4X54orYcIW2qMDs/6fFZwMJEJZsCIUxXey6VO8FgmEjpblCjJXoIaZ2Nk
Zw0fcjwfMqtMemIkcyJxc9jLNqBahWBREwFh0kEQxVY=
-----END CERTIFICATE-----


server.crt" 21L, 1269C
```

步骤4 清理中间过程文件，删除server.pem文件。

----结束

2.1.2 自制客户端的证书和密钥

您可以使用JDK自带的keytool工具自制服务端的密钥库文件（client.p12），再使用openssl工具将client.p12转换为client.pem，最后根据client.pem文件得到服务端的私钥文件（client.key）、证书文件（client.crt）。

步骤1 以Mind Studio的安装用户登录Mind Studio服务器。

步骤2 生成客户端的keystore文件“client.p12”。

在当前目录或切换到其它目录下，执行以下命令生成keystore文件。

1. 执行keytool命令。

```
keytool -genkey -storetype PKCS12 -keystore client.p12 -alias 1 -keysize 2048 -keyalg RSA -ext san=ip:xx.xx.xx.xx -validity 3650
```

在以上命令中，各参数的解释如下：

- **-genkey**表示生成一个钥匙对（公钥和私钥）。
- **-storetype PKCS12**表示指定密钥库的类型为PKCS12。
- **-keystore**参数后面跟着的是 keystore文件的名字，可修改。
- **-alias**参数后面跟着的是生成的密钥和证书的别名，可修改。
- **-keysize**参数后面跟着算法长度。
- **-keyalg**参数后面跟着生成密钥对的算法。
- **-ext san**参数用于绑定服务器的IP地址。ip需要设置为Device侧服务器的IP。*xx.xx.xx.xx*需要替换为实际Device侧服务器的IP。
- **-validity**参数后面跟着证书的有效期，以天为单位。如果不指定有效期，则默认有效期是90天。

2. 输入keytool命令后，回显信息如下，您需要根据提示信息输入。

   此处输入的密码建议与步骤2.2处设置的密码保持一致，输入的字符串长度大于等于6且小于等于15。

   ```
   Enter keystore password:（输入keystore的访问密码）
   Re-enter new password:（重复输入keystore的访问密码）
   ```

3. 输入密码后，您需要根据表2-2中的回显信息的提示，输入相应的信息。

   表 2-2 回显信息和输入信息

顺序	回显信息	输入信息
1	What is your first and last name?	这里必须是IDE-daemon-host侧服务器IP地址。 **说明** 如果Mind Studio与IDE-daemon-host合设在一台服务器上，此处IP地址可设置为“127.0.0.1”。
2	What is the name of your organizational unit?	根据待启用证书服务器实际所在的公司部门名称输入，例如“Huawei HISI”。
3	What is the name of your organization?	根据待启用证书服务器实际所在的公司名称输入，例如“Huawei”。
4	What is the name of your City or Locality?	根据待启用证书服务器实际所在的城市名输入，例如“HangZhou”。
5	What is the name of your State or Province?	根据待启用证书服务器实际所在的国家名称输入，例如“China”。
6	What is the two-letter country code for this unit?	根据待启用证书服务器实际所在的2位国家缩写代码输入，例如“CN”。

4. 输入完成后，系统将询问用户输入的信息是否正确，如果正确，请输入“y”或“yes”，否则请输入“n”或“no”。
5. 输入“y”或“yes”后，系统提示输入访问密钥的密码，根据提示输入密码。
6. 执行以上命令后，如果系统不提示错误，则可以在当前目录下查看生成的keystore文件“client.p12”。

步骤3 生成客户端的加密私钥client.key和证书client.crt。

1. 将“client.p12”文件转化成“client.pem”文件。

   ```
   openssl pkcs12 -in client.p12 -out client.pem
   ```

   在以上命令中，各参数的解释如下：

   - **-in**：该参数后面跟着的是pkcs12文件的路径，包含文件名。

- **-out**：该参数后面跟着的是证书或私钥文件的路径，包含文件名，文件名格式必须是*.pem。

openssl命令执行后，系统会回显如下信息，您需要根据提示输入密码，输入密码后，在执行openssl命令的目录下会生成client.pem文件。

```
Enter Import Password:
MAC verified OK
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
```

在“Enter Import Password”处输入访问密钥库文件**client.p12**的密码，与**步骤2.2**中输入的密码一致。

在“Enter PEM pass phrase”和“Enter PEM pass phrase”处输入访问**client.pem**的密码，与**步骤2.2**中输入的密码一致。

2. 创建一个空文件，命名为client.key。

   打开client.pem，复制“-----BEGIN ENCRYPTED PRIVATE KEY-----”到“-----END ENCRYPTED PRIVATE KEY-----”之间字符串（加密后的密钥），保存到client.key文件中。

```
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFDjBABgkqhkiG9w0BBQ0wMzAbBgkqhkiG9w0BBQwwDgQI06HoS+1lpmgCAg
MBQGCCqGSIb3DQMHBAjA2YuSpkitUASCBMhs2wO3njVstY0hJ7IibFriX9uq+E
nJebSi2gOxWq3NrTuzG/gthm1jvi/XW5P2Oq1DecFJZfYKnYB+6UWbM/N5zdBF
xnBxO7MAZv339mw7LsP4XydT3idCK7fUlhnqP9fv+bK6LpNCkpNGHrkXD3oyNF
bEHrGyoWJQ+WAHyn/bDGQW5sbi8mDXv1JR812bIYatRyu52qtm1aC8aJi4f5rJ
e6KunLt8uqj/Kxcz9TowLXKc55Rzwfn+7TjdIemQOuS3RJSz2eA1/edo6COMue
sOpFAK9Kz8Df0TE9G/NrX+wH+IPqKZt8IH7TECHFPHh1DtZA7dK7Hm6Dctr9dc
9KJZXEbwcSUXc1NqUefwLp1+B4OXxPT/7XvipZjvnWiljfyV6M02ZCLT3lSKlx
HAVcblvzS1ctJwuzTmBXQQBw/uj6FK82/LXS+CQbT579sGJ6m0vE3qRl6uEyZv
BZQxV26UX8Es4HJ7cJ/9InIkRNfkplhGSo8//JZFldXimRIVn3i8U8Cr10bRO4
GYvWJHoTzOi2DNsHG9G517fJBNIgSptCA4RLqCY96vv6ItSVqVuOGU+X0Tt+zA
skMC9/4t3qtK7n8Ej9zY4ekawmoi5QQrk0wWdGGF6yqCa9vzFr6xt20Ci43YmB
ZedDuRWK0lYzkZ+kwXijd9QDfwKZFptnSq8sFonhW8Z1xD6DDmz+NY+skE0RhB
5NTGs66egZ1B3ivmKa26qWDrm8IqgkHK+4VRC3qZsywFTOU2YpXGpS9x+hHIx8
EmrwlHMkRpH7YhWQ5xMwda+LvdIC504+AozUeKdj4KkNZgzf6F/3dUgp/moSg4
6ts0ZPyn16VpF/F8O4TTZuE43gY83v7Adivz6SEjn1d5+b6tgTv4Vn8bIfuWnC
maRiZTlrooGahsXUhRHLEbEpoH52jp7YGgUF6eI15qU1nJqDsAku6L8g5SZ2Uc
ZDTm11wwBajMi4U8kNvmwZ3BD8DXVK3ZClKTzB5XxW9aLmZ0CmmsUZfsHpB86B
CFbtNrMTMGDDWruO1ZeTI8qC3HV9ZbuDCAiuLR7ohveaFgQSXjHbHAmtx+H+O6
wl/aX8sITLoHa7GOmj4rY/5mfGw8HIjJFB8VIs6O5/nVEgKIVLAgpZCcLFcKaa
2b6vQ3f9ilqVTBMDn+eqRWFYPIg2pjz4upJNH4qazITTCZfmQYJ6TgrASmiOfM
IU3ch1oxDjUT8qhBAgYILHvp0wac0Kpn2kdBSDxjJ4ya2QPXd4BmaQPvJIPvPc
Qx6ZekhcxkjGeU8ZTduWZ6QNSIBisTokGx8yCpnIVUFVk3FVXFSmOFYMKmMRsP
egmZKxUwznokpkGOV+UCP4e39PdB8Ihk4i28SXpiy6Rz85rzRPaomlJUpG70Pc
1P3B/LG3lcC3ol+ta7F9S5lp6q+8FsGD3rkAz1OX4Hwg2ludDougpSPjJHB4jx
bIzZWSm+uckO/+7EBKTDa8zZYppXcLVHkevBayOPu+sVNPfRycEqKPLqX2nhpl
EyWpAfC7oCSD860yAv/rpw3WyigJ5JAY0yE1l/RdrB4fUwWfSZTfjhmajsNioP
VmM=
-----END ENCRYPTED PRIVATE KEY-----
~
~
~
~
"client.key" 30L, 1834C
```


3. 创建一个空文件，命名为client.crt。

   打开server.pem，复制“-----BEGIN CERTIFICATE-----”到“-----END CERTIFICATE-----”之间字符串（证书），保存到client.crt文件中。

```
-----BEGIN CERTIFICATE-----
MIIDcjCCAlqgAwIBAgIEZ+cJczANBgkqhkiG9w0BAQsFADBhMQswCQYDVQQGEwJD
TjELMAkGA1UECBMCemoxCzAJBgNVBAcTAmh6MQ8wDQYDVQQKEwZodWF3ZWkxDzAN
BgNVBAsTBmh1YXdlaTEWMBQGA1UEAxMNMTkyLjE2OC4xLjExMjAeFw0xOTAxMTYw
NDA5MTVaFw0xOTA0MTYwNDA5MTVaMGExCzAJBgNVBAYTAkNOMQswCQYDVQQIEwJ6
ajELMAkGA1UEBxMCaHoxDzANBgNVBAoTBmh1YXdlaTEPMA0GA1UECxMGaHVhd2Vp
MRYwFAYDVQQDEw0xOTIuMTY4LjEuMTEyMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8A
MIIBCgKCAQEAiLYU4m6XsSNLtoH8+IPu4PqaGvKZVw8siGp1wHr0S+aHKXT0EW9A
oodXqZbnxoeNayrK4p++SrVlMf3DgjnpICOocTReiMbVhYcC+rls9cQbOfOyBcxR
njBnQ1O0S8kP0l+s7E1yUsdf+jcCmroFFgCEGEf/L9H8rGRLhMIVz7p8lJbB89DZ
aEzRSfjCub20nfaqLgDcr+b+X8sz3AmaxRFjwHw2pvGPb8xUZ6fQ789HEM7ZJIHK
bhOjPxDu4O3G3EN6u0GtHc5sAzXfZ6Dt4DRFA60KGTMQ118N1EJ4SSPoywxWZRvJ
JeezQBsaFJQJMbKMilPCS70RRujDgNCW4QIDAQABozIwMDAPBgNVHREECDAGhwTA
qAFwMB0GA1UdDgQWBBTHkWqB1S135dvZcbr/pxM/OAsR/TANBgkqhkiG9w0BAQsF
AAOCAQEACW2t3VsLtv936GYYLwySzrnDIrMJMfrh74Xp4eeQHWojQb8volptO25Y
l85enPc6tzkwTohP1RtmDGOf+cshwfOMsao2R/OzzihAUjq9P7v2SUOZk4dprHR0
vUE4Iu/nD/DlQ+WZ58yeo8Ts8IL2MPl9MSigxYWoI7cuCLUS9o6HIwgy338eH7ZV
mucfhRwKfidJ+bEMCGfGmR+m1wfHLD3zceB77kLWKSSMIHWzD4LIDD2g2JEPdzps
F2qg4d6pc89LrYfDkN5AQ+ODNbKh1j0XkPfgVCj7a9Cm91b7qn2pt6bqVwZwfNT1
6f9gYP+0bcuumWG0XUN+/02WBU6Kbw==
-----END CERTIFICATE-----
~
~
~
~
~
~
"client.crt" 21L, 1257C
```

步骤4 制作信任证书密钥库。

1. 获取**2.1.1 自制服务端的证书和密钥**中生成的server.p12，复制server.p12，并将其重命名为catrust.jks。

```
cp server.p12 catrust.jks
```

2. 将客户端证书文件client.crt导入到catrust.jks密钥库文件中。

```
keytool -import -v -file client.crt -storetype PKCS12 -keystore catrust.jks
```

   在以上命令中，各参数的解释如下：

   - **-import**：将已签名的数字证书导入密钥库。
   - **-v**：显示密钥库中的证书详细信息。
   - **-file**：该参数后面跟着的client.crt是要导入的证书。
   - **-storetype PKCS12**表示指定密钥库的类型为PKCS12。
   - **-keystore**：该参数后面跟着的catrust.jks是密钥库的名称。

3. 按照提示输入访问密钥库catrust.jks（即server.p12）的密码，与**步骤2.2**中的密码一致。
4. 按照提示确认是否信任该证书，输入yes。
5. 查看是否信任了client.crt证书。

```
keytool -list -storetype PKCS12 -keystore  catrust.jks
```


在以上命令中，各参数的解释如下：

- **-list**：显示密钥库中的证书信息。
- **-storetype PKCS12**表示指定密钥库的类型为PKCS12。
- **-keystore**：该参数后面跟着的catrust.jks是密钥库的名称。

步骤5 导出CA证书（浏览器使用）

```
keytool -storetype PKCS12 -keystore catrust.jks -export -alias 1 -file CA.cer
```

在以上命令中，各参数的解释如下：

- **-export**：将别名指定的证书导出到文件中。
- **-alias**：参数后面跟着的*1*是别名。该参数的值需要与**步骤2.1**中设置的alias保持一致。
- **-storetype PKCS12**表示指定密钥库的类型为PKCS12。
- **-keystore**：该参数后面跟着的catrust.jks是密钥库的名称。

步骤6 生成IDE Daemon Host与Mind Studio的双向验证CA证书。

```
cat server.crt client.crt > ca.crt
```

步骤7 清理中间过程文件，删除client.pem文件。

----结束

3 向第三方机构申请证书（商用）

步骤1 客户自行向第三方机构提交证书申请文件。

第三方机构返回签名后的证书文件给客户，例如服务端证书server.crt、客户端证书client.crt。

步骤2 客户生成服务端的keystore文件“server.p12”。

```
openssl pkcs12 -export -clcerts -in server.crt -inkey server.key -out server.p12
```

步骤3 客户生成客户端的keystore文件“client.p12”。

```
openssl pkcs12 -export -clcerts -in client.crt -inkey client.key -out client.p12
```

步骤4 按照**2.1.1 自制服务端的证书和密钥**中的**步骤3**制作加密私钥server.key。

步骤5 按照**2.1.2 自制客户端的证书和密钥**中的**步骤3**~**步骤7**制作加密私钥client.key、catrust.jks、CA.cer、ca.crt文件。

----结束

4 替换证书

操作步骤

步骤1 以Mind Studio安装用户登录Mind Studio所在的服务器。

步骤2 切换到“~/tools/conf”目录下，创建存放Mind Studio新证书的目录，目录名称为“new_crts”,。

```
mkdir new_crts
```

说明

“~/tools”是默认的toolpath路径，该路径可在安装Mind Studio时由用户自定义，您可以在“scripts/env.conf”文件通过toolpath参数查看实际路径。您可以使用**find / -name 'env.conf'**命令查看script目录下的“env.conf”文件的位置。

步骤3 切换到“~/tools/bin”目录下，执行以下命令停止Mind Studio服务。

```
bash stop.sh
```

说明

如果是手动安装Mind Studio，则需要切换到scripts目录下执行以上命令。

步骤4 替换密钥证书文件。

- 对于自制的证书和密钥

  将**2.1.1 自制服务端的证书和密钥**中生成的证书密钥文件（server.crt\server.key \server.p12）、**2.1.2 自制客户端的证书和密钥**中生成的证书（client.crt\client.key \client.p12\catrust.jks\ca.crt）替换到“~/tools/conf/new_crts”路径下，用于替换的证书密钥文件名需要与“~/tools/conf/secure_keys”路径下的证书密钥文件名保持一致。

- 对于向第三方机构申请的证书

  将**3 向第三方机构申请证书（商用）**中生成的证书密钥文件（server.crt\server.key \server.p12\client.crt\client.key\client.p12\catrust.jks\ca.crt）

  替换到“~/tools/conf/new_crts”路径下，用于替换的证书密钥文件名需要与“~/tools/conf/secure_keys”路径下的证书密钥文件名保持一致。

步骤5 在“~/tools/scripts/crt.conf”配置文件中更换访问密钥库的密码。

1. 修改crt.conf配置文件中的“set_crtpass”参数值，将值改为访问密钥库的密码（明文字符串），与**步骤2.2**中设置的密码保持一致。
2. 切换到Mind Studio安装用户的家目录，例如：“/home/ascend”。

   依次执行如下命令，给Mind Studio安装用户添加sudo权限，命令中的*username*请替换为实际的Mind Studio安装用户的用户名。


```
su root
chmod +x add_sudo.sh
./add_sudo.sh username
```

执行完以上命令后，切换到Mind Studio安装用户下。

3. 切换到“tools/bin”目录下，执行以下命令，替换证书。

```
bash start.sh
```

a. 系统回显图4-1中的提示信息，输入“Y”或“y”，回车。

图 4-1 是否使用新证书

```
do you want use the new certifications?[Y/N]: y
```

b. 系统回显图4-2中的提示信息，输入“Y”或“y”，回车。

图 4-2 是否更换 IDE-daemon-host 下的证书

```
do you want to change the certifications to IDE-daemon-host?[Y/N]: y
```

c. 系统回显图4-3中的提示信息，输入IDE-daemon-host所在服务器的IP地址。

图 4-3 输入 IP 地址

```
please input the ide daemon host IP:
```

如果mindstudio和IDE Daemon host 安装在一台机器上需要输入内部IP 127.0.0.1

d. 若系统回显图4-4中的提示信息，则表示证书替换成功。

图 4-4 替换证书成功

```
change the all certifications done!!
now you can restart the mindstudio!
```

说明

- “~/tools/scripts/crt.conf”配置文件中的“set_crtpass”参数值若是明文字符串，在执行**bash stop.sh**命令后，系统会提示是否需要更换证书，如果确认更换证书，则“set_crtpass”参数值会被加密。
- 替换证书时，系统会自动用“~/tools/scripts/crt.conf”配置文件中的“set_crtpass”参数值（加密后的字符串）替换“~/tools/conf/profiler.cfg”文件中的“set_crtpass”参数值。
- 证书替换成功后，您可以**HwHiAiUser**用户登录Host侧服务器，在“~/ide_daemon”目录下查看IDE-daemon-host的证书。

4. 执行以下命令，停止Mind Studio服务。

```
bash stop.sh
```

5. 替换证书成功后，手动删除备份的文件夹。

以Mind Studio安装用户登录Mind Studio侧，删除“~/tools/conf”下的“backup_crt”目录。

以**HwHiAiUser**用户登录Host侧服务器，删除“~/ide_daemon”目录下的“backup_ide_daemon_crt”目录。

6. 执行以下命令，启动Mind Studio服务。

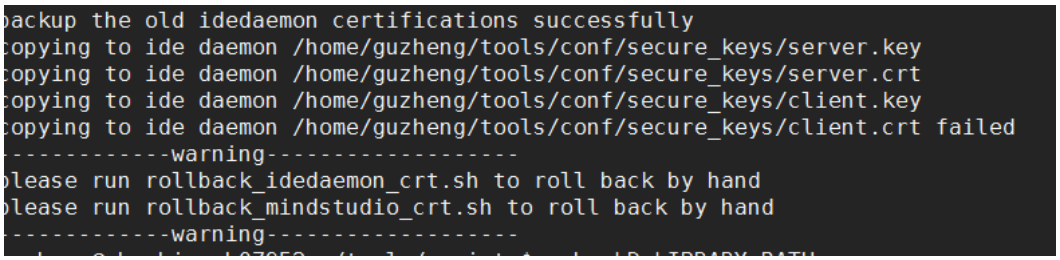
```
bash start.sh
```

----结束

异常处理

- 如果在替换证书过程中出现异常，导致程序退出，则可以重复**操作步骤**中的步骤重新替换证书。
- 如果在网络传输过程中IDE-daemon-host出现异常重启或断电，如**图4-5**所示，则您需要手动还原前一次的证书。

图 4-5 IDE-daemon-host 异常



还原证书的步骤如下：

a. 以Mind Studio安装用户登录Mind Studio服务器。

b. 执行以下命令，停止Mind Studio服务。

```
bash stop.sh
```

c. 在“~/tools/scripts”目录下执行如下命令还原Mind Studio（包括Profiling）的证书。

```
bash rollback_mindstudio_crt.sh
```

d. 执行以下命令，启动Mind Studio服务。

```
bash start.sh
```

e. 以**HwHiAiUser**用户登录Host侧服务器。

f. 在“~/ide_daemon”目录下执行如下命令还原IDE-daemon-host的证书。

```
bash rollback_idedaemon_crt.sh
```

g. 重启Host侧服务器。

```
reboot
```

执行reboot命令前，需要切换到root用户。

h. 确保Mind Studio与Host侧正常通信后，重复**操作步骤**中的步骤重新替换证书。

5 测试验证

步骤1 在Chrome浏览器的地址栏输入访问Mind Studio界面的地址。

步骤2 单击“了解详情”，可查看**图5-1**中的信息。

比对**2 获取证书**中生成的证书文件server.crt的内容与**图5-1**中证书信息是否一致，如果一致，则表示证书替换成功；若不一致，请执行**4 替换证书**中的步骤重新替换证书。

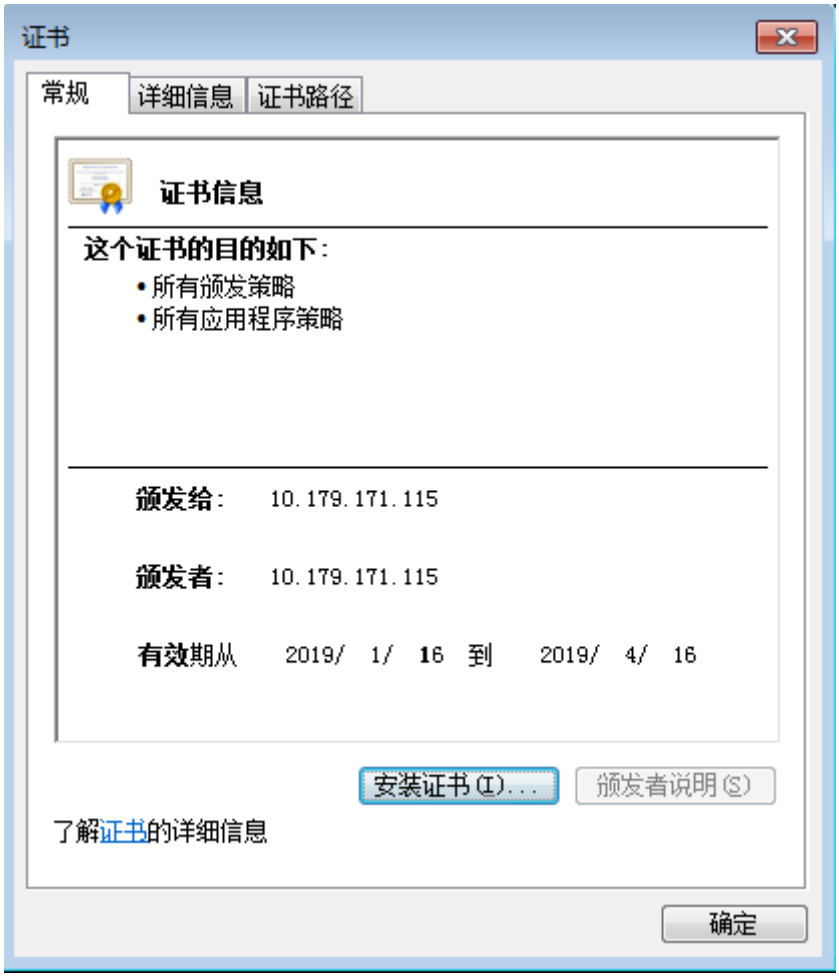
图 5-1 证书信息

步骤3 单击“高级”，进入Mind Studio界面。

步骤4 清除不安全链接。

1. 关闭Chrome浏览器的所有窗口。
2. 将**2 获取证书**中生成的CA.cer文件存放到浏览器所在的机器。
3. 双击CA.cer文件后，单击“打开”。

4. 单击“安装证书”。



5. 按照向导提示，单击“下一步”。
6. 选中“将所有的证书放入下列存储”，并单击“浏览”，选择“受信任的根证书颁发机构”。

7. 按照向导提示，依次单击“下一步”、“完成”。

步骤5 重新打开Google浏览器，输入访问Mind Studio界面的网址，证书认证成功，不安全标志消失。

----结束

