

中華電信 HiPKI 憑證管理中心 (OVTLSA)

TOMCAT 伺服器 SSL 憑證請求檔製作與憑證安裝手冊

聲明：本手冊之智慧財產權為中華電信股份有限公司（以下簡稱本公司）所有，本公司保留所有權利。本手冊所敘述的程序係將本公司安裝相關軟體的經驗分享供申請 SSL 伺服器軟體憑證用戶參考，若因參考本手冊所敘述的程序而引起的任何損害，本公司不負任何損害賠償責任。

本手冊的申請程序，已經在 Windows 系統 + Tomcat 7.0 版測試過，您所使用的版本或環境可能與本手冊所測試的版本有所差異，若是如此則請參考您的 Tomcat 相關使用手冊，適度調整申請步驟。

目錄

Tomcat SSL 憑證請求檔製作手冊.....	2
Tomcat SSL 憑證安裝操作手冊.....	5
附件一：停用 SSLv3.0.....	8

Tomcat SSL 憑證請求檔製作手冊

一、 製作憑證請求檔

1. 開始前，請先確認您 Java 的版本

由於 Tomcat 的底層是 Java，如果其所使用的 Java(JDK)版本是 1.5 版以前的版本，將無法安裝 RSA 4096 位元金鑰長度的憑證，因為舊版的 Java 最多只支援 RSA 2048 bits 的金鑰長度，這將造成 Java Keystore 不會將根憑證、中繼憑證及 SSL 憑證視為 1 個憑證串鏈，結果在 SSL Handshake 中，中繼憑證就不會被送到 Client 端，建議請使用最新版本 Java(JDK)版本。

2. 開啟命令提示字元。

3. 在 %JAVA_HOME%\bin 目錄下，請執行

keytool -genkey -alias <金鑰的 alias name> -keyalg RSA -keysize 2048 -keystore <keystore 儲存路徑>(請自行輸入需要的路徑與檔名)。

- 若您非第 1 次申請憑證，請確認您所指定的路徑與檔名不會覆蓋線上正在使用的憑證。
- 此指令會在指定目錄下產生".keystore"檔(內含私密金鑰)，請勿於提出憑證申請後重複執行此指令，否則舊的".keystore"檔將會被覆蓋。
- 依照國際密碼學之規範，2014 年起不要再使用 RSA 1024 位元之憑證，請產製 RSA 2048 位元(含)以上金鑰長度的金鑰對。
- 請妥善保管此".keystore"檔。

```
C:\Program Files\Java\jdk1.7.0_17\bin>keytool -genkey -alias tomcat -keyalg RSA
-keysize 2048 -keystore D:\.keystore
輸入金鑰儲存庫密碼:
重新輸入新密碼:
您的名字與姓氏為何?
  [Unknown]: www.test.com.tw
您的組織單位名稱為何?
  [Unknown]: 政府網路處
您的組織名稱為何?
  [Unknown]: 中華電信股份有限公司數據分公司
您所在的城市或地區名稱為何?
  [Unknown]: Taipei
您所在的州及省份名稱為何?
  [Unknown]:
此單位的兩個字母國別代碼為何?
  [Unknown]: TW
CN=www.test.com.tw, OU=政府網路處, O=中華電信股份有限公司數據分公司, L=Taipei, S
T=Unknown, C=TW 正確嗎?
 [否]: Y
輸入 <tomcat> 的金鑰密碼
  <RETURN 如果和金鑰儲存庫密碼相同>:
```

4. 出現「輸入 keystore(金鑰儲存庫)密碼」：請輸入一個密碼，用以保護此儲存庫(請妥善保存此組密碼)。
5. 接著依照畫面填入所需資料：

您的名字與姓氏為何？	網站名稱(ex: www.test.com.tw)
您的組織單位名稱為何？	單位名稱(ex: IT)
您的組織名稱為何？	組織名稱(ex: CHT)
您所在的城市或地區名稱為何？	城市(ex: Taipei)
您所在的州及省份名稱為何？	可不填，按 Enter 跳過
此單位的兩個字母國別代碼為何？	填入 TW
6. 檢查所輸入的資料是否正確，若正確，請輸入 Y。
7. 出現「輸入 <tomcat> 的金鑰密碼」：請直接按"Enter"鍵。(注意：建議此步驟所設的密碼與金鑰儲存庫相同，否則 Tomcat 可能無法使用此金鑰來啟動 SSL)。
8. 在 %JAVA_HOME%\bin 下，執行
`keytool -certreq -alias <上一步驟所用的 alias name> -file <憑證請求檔儲存路徑> -keystore <keystore 檔案所在路徑>`

```
C:\Program Files\Java\jdk1.7.0_17\bin>keytool -certreq -alias tomcat -file D:\certreq.txt -keystore D:\keystore
```

輸入金鑰儲存庫密碼：

- 二、 此時憑證請求檔(certreq.txt)製作完成，使用憑證請求檔至中華電信公開金鑰基礎建設服務網站 (<https://chtca.hinet.net/>) 依照網頁說明申請 SSL 憑證。
- 若屬於中華電信公司各單位申請 SSL 憑證者，請從企業入口網站電子表單之資訊表單 IS14-伺服器應用軟體憑證申請/異動單提出申請。

補充說明 1: 中華電信公開金鑰基礎建設服務之程式會擷取憑證請求檔中的公開金鑰，但不會使用憑證請求檔中於上圖所輸入之資訊，而是以於申請網頁上所填入的組織資訊與完全吻合網域名稱(Fully Qualified Domain Name, FQDN)為準，並記載於所簽發的 SSL 憑證裡面的欄位[如憑證主體名稱(Subject Name)之一般名稱(Common Name)或憑證主體別名(Subject Alternative Name)等欄位]。

補充說明 2: 若您是申請多網域 SSL 憑證或萬用網域 SSL 憑證，僅需要產生 1 個憑證請求檔(產生憑證請求檔之過程就是幫您的伺服器產製 1 對金鑰對，私密金鑰與密碼由伺服器管理者保管，公開金鑰會包含在憑證請求檔內，憑證管理中心審驗您的身分與網域名稱擁有權或控制權後，所簽發的憑證會記載申請者的組織資訊、完全吻合網域名稱與公開金鑰在 SSL 憑證內。後續先安裝 SSL 憑證串鍊於產生憑證請求檔之站台，再將私密金鑰與憑證備份後匯入其他站台，不同廠牌伺服器之匯出與匯入可參考手冊或寫電子郵件給本管

理中心技術客服信箱 caservice@cht.com.tw 詢問，不需要每個網站站台都分別產生憑證請求檔。

Tomcat SSL 憑證安裝操作手冊

一、 下載憑證串鏈，包含 4 張憑證，分別是(1)eCA 根憑證(ePKI Root CA 憑證，也就是中華電信憑證總管理中心自簽憑證)、(2)eCA to HRCA 交互憑證(eCA 簽發給 HRCA 之交互憑證)、(3)HiPKI OV TLS CA 中繼憑證(中華電信 HiPKI OV TLS 憑證管理中心自身憑證)與(4)OV TLS CA 簽發給用戶的 SSL 伺服器憑證，可採以下兩種方式之一取得：

1. 您若是本公司之客戶，技術聯絡人的電子郵件信箱會收到憑證串鏈壓縮檔，解壓縮後包括 4 個檔案，分別是 eCA 根憑證(檔名為 ROOTeCA_64.crt)、eCA to HRCA 交互憑證(檔名為 eCA1-to-HRCA1.crt)、OV TL SCA 中繼憑證(檔名為 OVTLSCA1_b64.crt)與用戶端 SSL 伺服器軟體憑證(檔名為 32 個英數字所組成，此為憑證序號)。

若是中華電信之所屬單位，於經審驗核准申請之電子表單的資訊表單「IS 14-伺服器應用軟體憑證申請/異動單」頁面下方，可以下載憑證串鏈壓縮檔，解壓縮後可以取得憑證串鏈 4 個檔案。

2. 從網站查詢與下載：

eCA 憑證：

https://eca.hinet.net/download/ROOTeCA_64.crt

eCA to HRCA 憑證：

<https://eca.hinet.net/download/eCA1-to-HRCA1.crt>

HiPKI OV TLS CA 憑證：

https://eca.hinet.net/repository-h/download/OVTLSCA1_b64.crt

SSL 憑證下載：您若是本公司之客戶，請至 CHTCA 網站點選「TLS 憑證效期查詢及下載」，進行 SSL 憑證下載。

若您是中華電信之員工，負責管理單位之伺服器，請至 <https://chtra.cht.com.tw/> 點選「憑證與卡片作業」，再點選「憑證查詢」，下載 SSL 憑證。

二、 SSL 憑證安裝，請使用您之前產生憑證請求檔的 Keystore 來執行匯入動作 (依信任關係，由最上層憑證，依序往下安裝)

1. 安裝根憑證。

在 %JAVA_HOME%\bin 目錄下執行

```
keytool -import -alias eCA -file D:\ROOTeCA_64.crt -keystore  
D:\.keystore
```

- 請依實際您存放檔案的位置調整指令。
- 待出現 Enter keystore password：請輸入密碼。

- 待出現 Trust this certificate：請輸 yes。
2. 安裝交互憑證。

在 %JAVA_HOME%\bin 目錄下執行

```
keytool -import -alias eCAtoHRCA -file D:\eCA1-to-HRCA1.crt -keystore D:\.keystore
```

 - 請依實際您存放檔案的位置調整指令。
 - 待出現 Enter keystore password：請輸入密碼。
 3. 安裝中繼憑證。

在 %JAVA_HOME%\bin 目錄下執行

```
keytool -import -alias OVTLSCA1 -file D:\OVTLSCA1_b64.crt -keystore D:\.keystore
```

 - 請依實際您存放檔案的位置調整指令。
 - 待出現 Enter keystore password：請輸入密碼。
 4. 確認 PrivateKeyEntry 的 alias name

在 %JAVA_HOME%\bin 目錄下執行

```
keytool -list -keystore D:\.keystore
```

 - 請依實際您存放檔案的位置調整指令。
 - 待出現 Enter keystore password：請輸入密碼。
 - 找到 PrivateKeyEntry 對應的 alias name，**範例為 tomcat**
 - 若您的 keystore 沒有 PrivateKeyEntry，放入 server 後，SSL 也無法成功連線。請找出原 keystore 檔案，或是重新申請。

```
C:\>keytool -list -keystore D:\.keystore
Enter keystore password:
Keystore type: JKS
Keystore provider: SUN

Your keystore contains 4 entries

eca, 2024年1月17日, trustedCertEntry,
Certificate fingerprint (SHA-256): C0:A6:F4:DC:63:A2:4B:FD:CF:54:EF:2A:6A:08:2A:0A:72:DE:35:80:3E:2F:F5:FF:52:7A:B5:D8:7
2:06:DF:D5
tomcat, 2024年1月17日, PrivateKeyEntry,
Certificate fingerprint (SHA-256): 19:C9:5C:DA:C3:23:09:F5:97:A0:F8:C5:BD:51:EB:67:7F:7D:96:8F:67:75:EF:22:1F:99:B0:2C:4
9:C2:AE:CE
ovtlscal, 2024年1月17日, trustedCertEntry,
Certificate fingerprint (SHA-256): D3:4A:5B:98:1A:85:CA:07:5D:B6:2C:BA:C4:15:EF:65:9D:95:33:90:40:CA:47:68:68:62:5D:4A:A
2:3A:98:49
ecatohrca, 2024年1月17日, trustedCertEntry,
Certificate fingerprint (SHA-256): 68:07:C9:72:35:C5:EC:60:90:26:9A:4B:5F:ED:FA:B4:69:86:E4:2F:4D:67:D2:ED:DD:CF:6E:45:C
F:0D:FA:80

Warning:
The JKS keystore uses a proprietary format. It is recommended to migrate to PKCS12 which is an industry standard format
using "keytool -importkeystore -srckeystore D:\.keystore -destkeystore D:\.keystore -deststoretype pkcs12".

C:\>
```

5. 匯入 SSL 伺服器應用軟體憑證。

在 %JAVA_HOME%\bin 目錄下執行

```
keytool -import -alias tomcat -file D:\5A4A7FB6FE24F6FFAC50A623568C6E9F.cer -keystore D:\.keystore
```

- 請依實際您存放檔案的位置調整指令。
 - 待出現 Enter keystore password：請輸入密碼。
6. 修改 Tomcat server.xml 設定
- 開啟 %tomcat_HOME%\conf\server.xml
 - 找到如下圖的地方，修改(加入) keystoreFile、keystorePass 的參數

```

server.xml - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

        redirectPort="8443" />
<!-- A "Connector" using the shared thread pool-->
<!--
<Connector executor="tomcatThreadPool"
            port="8080" protocol="HTTP/1.1"
            connectionTimeout="20000"
            redirectPort="8443" />
-->
<!-- Define a SSL HTTP/1.1 Connector on port 8443
This connector uses the JSSE configuration, when using APR, the
connector should be using the OpenSSL style configuration
described in the APR documentation -->
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
            maxThreads="150" scheme="https" secure="true"
            keystoreFile="D:\.keystore" keystorePass="your password"
            clientAuth="false" sslProtocol="TLS" />

<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />

<!-- An Engine represents the entry point (within Catalina) that processes
every request. The Engine implementation for Tomcat stand alone
analyzes the HTTP headers included with the request, and passes them
on to the appropriate Host (virtual host).
Documentation at /docs/config/engine.html -->

```

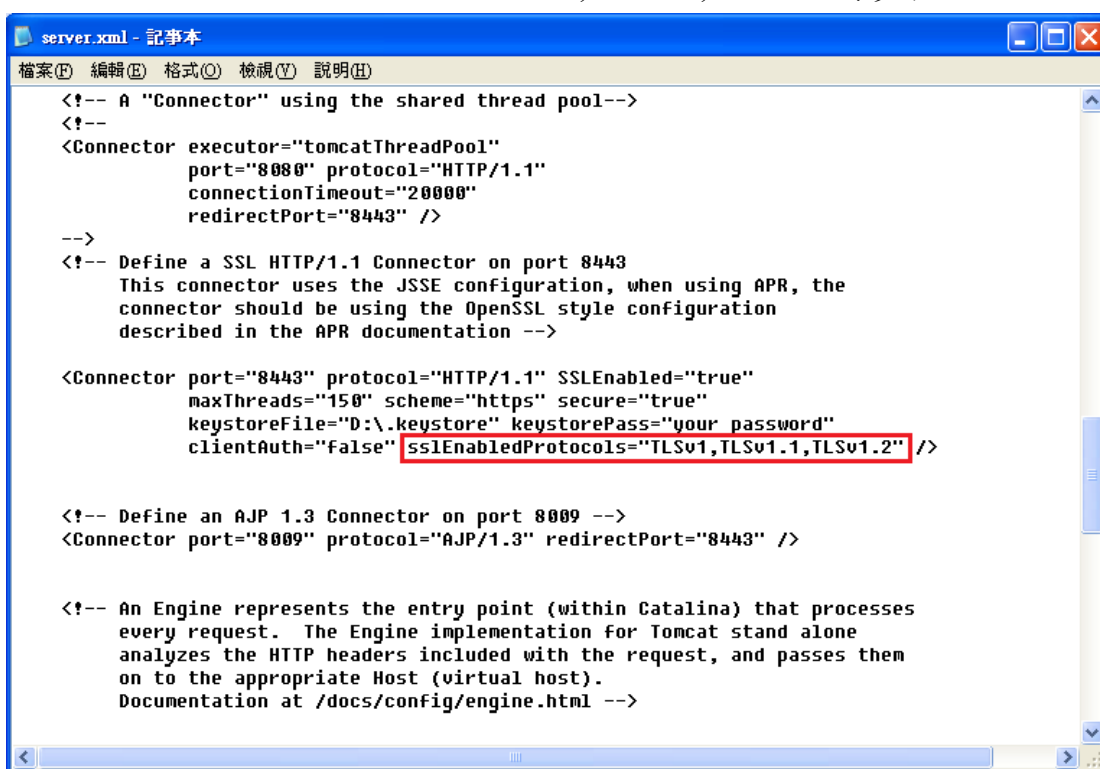
7. 重新啟動 Tomcat
8. 成功後，請以 https 連線試試加密通道。(注意：Tomcat 預設 https 為 8443 port，如需要 443 port，請自行修改)
9. 依照您的網路架構，您可能需要於防火牆開啟對應 https 的 port。
- 三、 安裝 SSL 安全認證標章：

請用戶參考技術聯絡人的電子郵件信箱所收到 SSL 憑證串鏈電子郵件內文的 SSL 安全認證標章安裝說明，將網站 SSL 安全認證標章安裝成功，網友可瀏覽您所維護網站所安裝 SSL 憑證的狀態與資訊。

請中華電信公司負責維護網站的同仁，參考從企業入口網站的電子表單之資訊表單「IS14-伺服器應用軟體憑證申請/異動單」所下載 SSL 憑證串鏈檔案中的 SSLSealispec.txt，將網站 SSL 安全認證標章安裝成功。

附件一：停用 SSLv3.0

- 開啟 %tomcat_HOME%\conf\server.xml
- 找到如下圖的地方，修改(加入)
 - 若您使用的 Tomcat 版本為 5 或 6(6.0.38 以前)
 - ◆ sslProtocols="TLSv1,TLSv1.1,TLSv1.2"的參數
 - 若您使用的 Tomcat 版本為 6(6.0.38 以後)或 7
 - ◆ sslEnabledProtocols="TLSv1,TLSv1.1,TLSv1.2"的參數



```
<!-- A "Connector" using the shared thread pool-->
<!--
<Connector executor="tomcatThreadPool"
    port="8080" protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="8443" />

-->
<!-- Define a SSL HTTP/1.1 Connector on port 8443
    This connector uses the JSSE configuration, when using APR, the
    connector should be using the OpenSSL style configuration
    described in the APR documentation -->

<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
    maxThreads="150" scheme="https" secure="true"
    keystoreFile="D:\.keystore" keystorePass="your password"
    clientAuth="false" sslEnabledProtocols="TLSv1,TLSv1.1,TLSv1.2" />

<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />

<!-- An Engine represents the entry point (within Catalina) that processes
    every request. The Engine implementation for Tomcat stand alone
    analyzes the HTTP headers included with the request, and passes them
    on to the appropriate Host (virtual host).
    Documentation at /docs/config/engine.html -->
```

- 重新啟動 tomcat，使用可以測試工具（註 1、註 2）進行檢測，看 SSL3.0 是否已停用。

註 1: 例如行政院國家資通安全會報技服中心網頁

<http://www.icst.org.tw/NewInfoDetail.aspx?seq=1436&lang=zh> 有介紹兩種檢測伺服器端 SSL 協定的工具：(1) TestSSLServer

(<http://www.bolet.org/TestSSLServer/>) (2) QUALYS SSL LABS SSL Server Test 檢測工具(<https://www.ssllabs.com/ssltest/index.html>, 也是 CA/Browser Forum 網站建議的檢測工具)可偵測伺服器所使用之加密協定，因 2014 年 10 月中國際公告了 SSLv3 加密協定存在中間人攻擊弱點，弱點編號 CVE-2014-3566 (POODLE)，故建議不要使用 SSL V3 協定，請改用 TLS 最新協定。

註 2:

(1) 若是用戶端各平台之瀏覽器要停止使用 SSL V3 協定可參考

<https://zmap.io/sslv3/browsers.html> 之英文說明

(2) 請超連結至 <https://dev.ssllabs.com/ssltest/viewMyClient.html> 可檢測您用戶端之瀏覽器是否已經停用 SSL V3。

(3) 若是 I.E. 瀏覽器可於工具列-> 網際網路選項->進階->安全性取消勾選使用 SSL V3 與使用 SSL V2，或參考下圖設定（取材自行政院國家資通安全會報技服中心網頁

<http://www.icst.org.tw/NewInfoDetail.aspx?seq=1436&lang=zh>)

