112年度政府機關網站憑證應用教育訓練

112年8月

大綱

1 **HTTPS**安全連線說明及相關安全設定

2 **GTLSCA ACME Certbot**

議程

項次	講題	時間
1	網站導入https及相關安全設定	45分鐘
2	休息時間	10分鐘
3	GTLSCA ACME Certbot	35分鐘
4	QA&意見交流時間	30分鐘

1. HTTPS安全連線說明及相關安全設定

HTTPS安全連線說明及相關安全設定

1.1. 協定說明

1.2. HTTPS設定注意事項

1.3. TLS類憑證申請說明

1.4. TLS類憑證安裝說明

1.5. TLS類憑證常見問題

1.6. 其他建議

1.1. 協定說明

HTTPS協定說明

1. 為確保政府網站傳輸之安全性，各政府機關建置、主責之網站應全面導入安全通訊協定(以下稱HTTPS)。HTTPS使用TLS憑證加密封包，降低資料傳輸遭竊取之風險。
2. 為便利各**政府機關**申請TLS憑證，數位部建置政府伺服器數位憑證管理中心(GTLSCA)，專職簽發TLS憑證供各機關申請使用，效期為1年，已申請者應於效期前續約。

HTTPS協定說明

HTTPS可以提供多種安全保障，包括：

- 保護使用者隱私：加密網頁連線，防止駭客竊取個人資訊。
- 提升網站安全性：防止駭客竊取網站的敏感資訊。(例:帳密)
- 提高網站排名：使用HTTPS的網站在搜尋排名較前。
- 增加使用者信任：使用HTTPS的網站會在網址列中顯示綠色鎖頭圖示，顯示網站是安全的。

1.2. HTTPS設定注意事項

HTTPS設定注意事項

如貴機關網站已使用**HTTPS**，仍需檢查是否有同時使用**HTTP**，建議將 **HTTP** 關閉或將流量重新導向至**HTTPS**，設定網站之**redirect Port**其屬性設定指向正確埠**(443)**，建議以下兩項設定搭配使用：

(1)**使用301/302轉址**

(2)**使用HSTS設定(建議設定1年)**

使用TLS 1.2以上版本

(1)使用301/302轉址

1. 透過設定301/302轉址，讓HTTP網站重新導向至HTTPS。
2. 301轉址(建議)：(永久性轉址)

   將舊網址永久轉導向新網站，並將頁面權重導向新網站。

3. 302轉址：(暫時性轉址)

   僅暫時轉導向新網站，且不移轉頁面權重。

(2)HSTS 設定

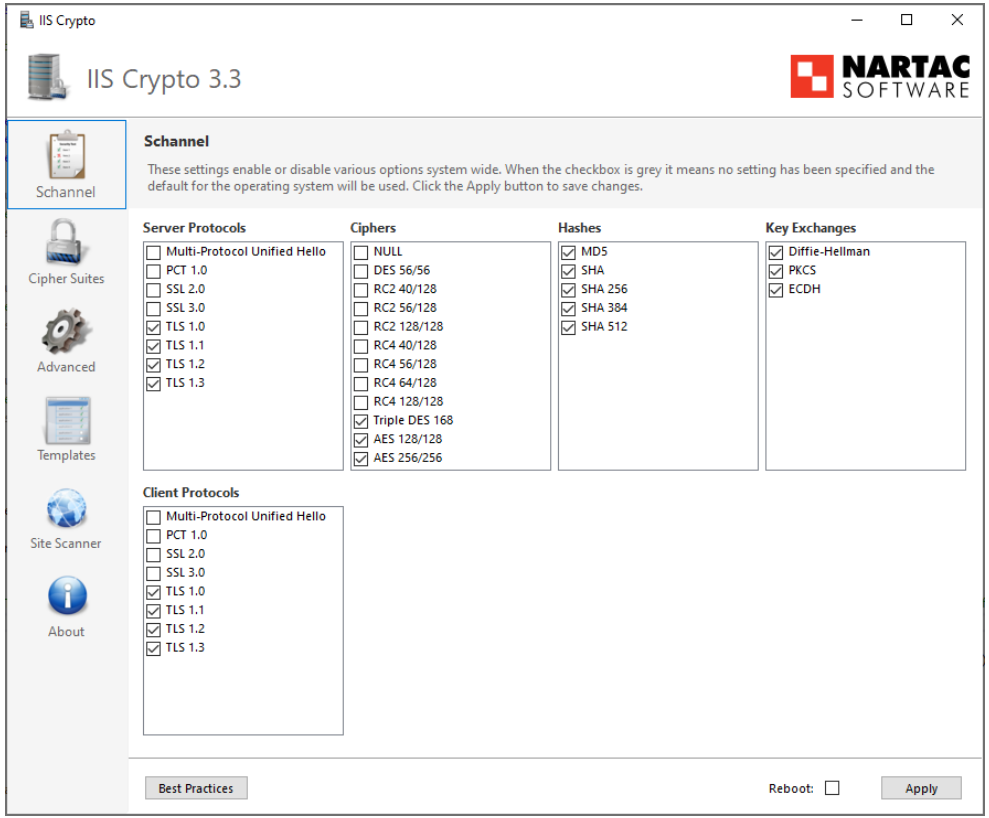
1. HSTS說明：
   讓用戶強制使用HTTPS與網站進行連線。
2. 設定方式：
   - 在HTTP header 加入 Strict-Transport-Security 。
   - 參數max-age(瀏覽器記得網站使用HTTPS連線之時間)，建議設定為 1年。

TLS連線演算法版本建議

- 說明
  - 建議使用TLS 1.3, TLS 1.2或更新版本的連線演算法。
- 版本說明
  - TLS 1.2支援的加密演算法包括AES-GCM、AES-CBC、3DES和RC4等，其中AES-GCM被認為是最安全的加密演算法之一。
  - TLS 1.3進一步強化了安全性，移除不安全加密演算法，僅保留AES-GCM和CHACHA20-POLY1305等安全的演算法。

TLS連線演算法版本設定-IIS

- 手動更改Regedit(很複雜)
  - http://www.waynezim.com/2011/03/how-to-disable-weak-ssl-protocols-and-ciphers-in-iis/
  - https://www.sslshopper.com/article-how-to-disable-ssl-2.0-in-iis-7.html
- 可使用工具IISCrypto輔助。
  - https://www.nartac.com/Products/IISCrypto



TLS連線演算法版本設定-apache

- 把禁用連線演算法字串寫到 **/etc/httpd/conf.d/ssl.conf** 裡(Windows版本路徑不同)，存檔後重新啟動**apache**服務即可。

```
SSLProtocol all -SSLv2 -SSLv3
SSLCipherSuite ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-AES128-
GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-
SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-
RSA-AES128-SHA256:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA:ECDHE-
ECDSA-AES256-SHA384:ECDHE-ECDSA-AES256-SHA:ECDHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-
AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES256-SHA:ECDHE-ECDSA-DES-CBC3-SHA:ECDHE-RSA-DES-
CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES128-GCM-SHA256:AES256-GCM-SHA384:AES128-SHA256:AES256-
SHA256:AES128-SHA:AES256-SHA:DES-CBC3-SHA:!DSS
SSLHonorCipherOrder on
SSLCompression off
SSLSessionTickets off

# OCSP Stapling, only in httpd 2.3.3 and later
SSLUseStapling on
SSLStaplingResponderTimeout 5
SSLStaplingReturnResponderErrors off
SSLStaplingCache shmcb:/var/run/ocsp(128000)
```

參考:

[1] https://www.sslshopper.com/article-how-to-disable-weak-ciphers-and-ssl-2.0-in-apache.html

[2] https://httpd.apache.org/docs/2.0/mod/mod_ssl.html#sslciphersuite

[3] https://apache.tutorials24x7.com/blog/how-to-enable-tls-1-2-and-tls-1-3-in-apache-web-server

TLS連線演算法版本設定-JavaBased Server

- 把禁用連線演算法字串寫到 server.xml裡，存檔後重新啟動JavaBased Server服務即可。

```
<!-- JSSE Connector -->
<Connector port="8443" protocol="org.apache.coyote.http11.Http11Protocol"
           maxThreads="150" SSLEnabled="true" scheme="https" secure="true"
           clientAuth="false" sslProtocol="TLS" sslEnabledProtocols="TLSv1.2,TLSv1.1,TLSv1"
           keystoreFile="/path/to/keystore.jks"
           keystorePass="keystorepasswordhere"
           ciphers="TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA,
                    TLS_RSA_WITH_AES_128_CBC_SHA,
                    TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,
                    TLS_RSA_WITH_AES_128_CBC_SHA256,
                    TLS_RSA_WITH_AES_128_GCM_SHA256,
                    TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,
                    TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,
                    TLS_RSA_WITH_AES_256_CBC_SHA,
                    TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA,
                    TLS_RSA_WITH_AES_256_CBC_SHA256,
                    TLS_RSA_WITH_AES_256_GCM_SHA384,
                    TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,
                    TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384"
  />
```

參考:

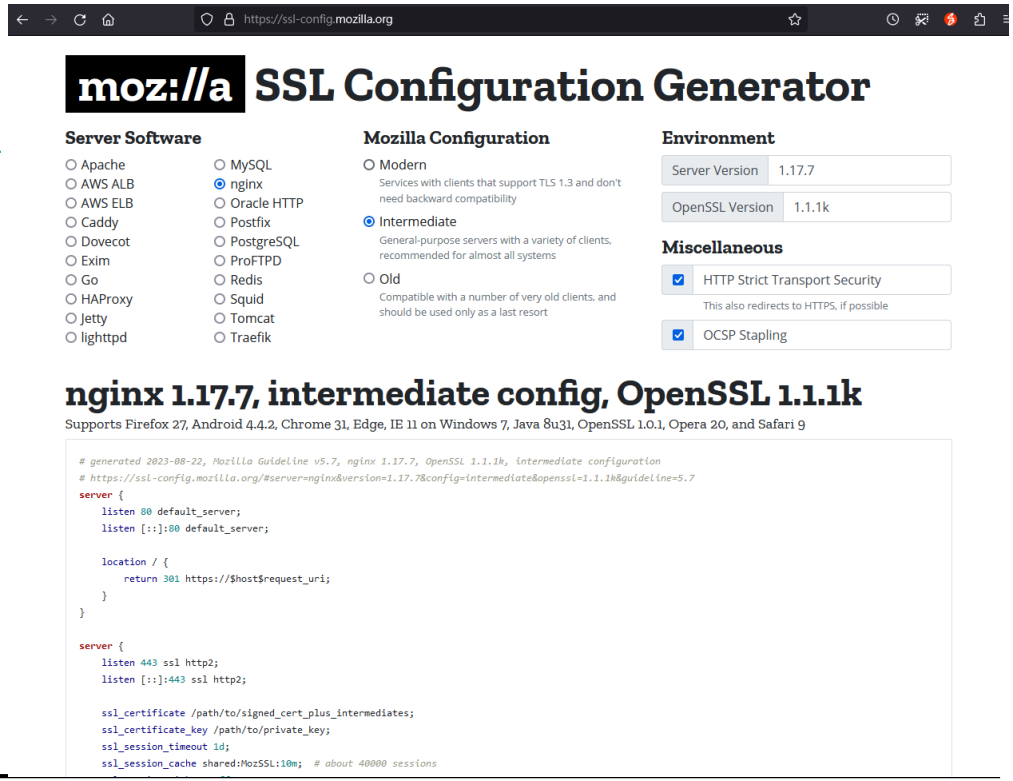
[1] https://tomcat.apache.org/tomcat-8.5-doc/config/http.html#SSL_Support

[2] https://grok.lsu.edu/article.aspx?articleid=17596

[3] https://stackoverflow.com/questions/23762762/tomcat-tlsv1-2-with-strong-ciphers-not-working

SSL Configuration Generator

- https://ssl-config.mozilla.org/



1.3. TLS類憑證申請說明

申請TLS憑證步驟

- 申請TLS憑證的兩種方式：
  - 發文申請。(約需2-3個工作天)
  - 使用GCA IC卡。(較快:建議使用)
- 申請流程請參考GTLSCA網站說明(https://gov.tw/iNH)
- 至OID網站查詢機關(構)之單位識別碼(OID)，如無則須申請。
- 至GTLSCA網站選擇TLS類憑證申請，線上填寫申請表，並製作憑證請求檔(CSR檔)，完成後上傳申請資料。

憑證請求檔(CSR)產製

- SSL憑證請求檔(CSR)製作與憑證安裝手冊可至(https://gov.tw/25A)下載。
- 上述網址包含Microsoft IIS、Apache、Tomcat網站伺服器的安裝手冊

憑證請求檔(CSR)產製(1/2)

1. IIS
   - 利用IIS管理員產製
   - 私密金鑰會自動產生在IIS主機內，可利用mmc工具匯出備份
2. Apache (Nginx)
   - 使用OpenSSL產製
   - 產生的server.key檔案即為私密金鑰，建議備份避免遺失
3. Tomcat (JBoss、WebLogic等JAVA Based Web Server)
   - 使用JAVA keytool產製
   - 產生的.keystore檔案內含私密金鑰，建議備份避免遺失

憑證請求檔(CSR)產製 (2/2)

憑證請求檔內容

(Certificate Signing Request, CSR)

憑證請求 CSR：

Download Copy

```
-----BEGIN CERTIFICATE REQUEST-----
MIICrjCCAZYCAQAwaTELMAkGA1UEBhMCVFcxDzANBgNVBAgMBlRhaXdhbj
EPMA0G
A1UEBwwGVGFpcGVpMRYwFAYDVQQKDA1DbG91ZG1heCBJbmMuMRMwE
QYDVQQDDAph
YWEuZ292LnR3MQswCQYDVQQLDAJJVDCCASIwDQYJKoZIhvcNAQEBBQA
DggEPADCC
AQoCggEBALRISNLNhKU/o0M/ZilOJtqEHntYYx4+4QlLh7Px+i4OcXYkvjxru3g0
sJBMESRLmuWpJFp7eJmQ2SCR8PWJeBDkQLfmj5+4nTp1/oD7VfMg3YdYttZz
NgK8
EJlmY7HbffRL4bA6VjFGYwRo2rNVlPY2MXt9alXwy2apQg4kTPrL7Tn833lcwZ1
```

憑證私鑰 KEY：

Download Copy

```
-----BEGIN PRIVATE KEY-----
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBKcwggSjAgEAAoIBAQC0SEjSzYSl
P6ND
P2YiDibahB57WGMePuEJS4ez8fouDnF2JL48a7t4NLCQTBEkS5rlqSRae3iZkNk
g
kfD1iXgQ5EC35o+fuJ06df6A+1XzlN2HWLbWczYCvBCSJmOx2330S+GwOlYxR
mME
aNqzVZT2NjF7fWpV8MtmqUIOJEz6y+05/N95XMGdRGZ0bFzd5n5MmJ+Ae270
sAFp
oK6CbFQWuDVfpPa6UgVjcju4alX93xc86q4GaAnlF+Oq8P79pOHTD8H8nGbge
Qwj
```

憑證相關檔案格式介紹-1/2

常見副檔名	檔案內容	用途說明
.cer/.crt	X.509 憑證 (RFC 5280)	常用於憑證 (不含私密金鑰)
.pfx	PKCS#12 憑證/金鑰封裝檔	金鑰匯入/匯出、備份，或是在不同應用伺服器交換使用 (如IIS轉至Tomcat)
.p7b	PKCS#7 多張憑證封包	將多張憑證封於單一檔案，常用於存放CA憑證串鍊
.jks	Java金鑰儲存檔	以Java keytool工具產製的金鑰/憑證儲存檔

憑證相關檔案格式介紹-2/2

常見副檔名	檔案內容	用途說明
.pem	X509憑證與公鑰，Base64格式	透過ACME Certbot申請，CA核發成功後取得，作為憑證檔安裝到AP伺服器檔案。
.key	私密金鑰，Base64格式	產製金鑰對、憑證請求檔時，產於本機上，不建議離開同一台機器，以便保證連現安全。

1.4. TLS類憑證安裝說明

如何安裝TLS憑證

- 憑證簽發後，系統將以電子郵件發送「憑證接受通知信」，請依照通知信說明接受憑證。
- 完成憑證接受後請至GTLSCA網站之「憑證查詢及下載」下載該憑證。
- 須至GTLSCA下載憑證串鍊中其他相關憑證：
  - 憑證串鍊需完整安裝(不可僅安裝TLS憑證)
  - GTLSCA SSL之憑證串鍊為：eCA → eCA1_to_eCA2-New → GTLSCA →用戶TLS憑證
  - 如憑證串鍊設定錯誤，某些瀏覽器將出現不信任警告。(檢測網站SSL Checker)

連線演算法設定檢查清單

- TLS憑證有效性
  - 憑證鏈檢查
  - CRL檢查/OCSP檢查
- 正確的公私鑰對
- 正確的憑證格式
- 合乎時宜的加密演算法
- 安全通訊協定的設置
- 強制HTTPS連線
  - 301/302轉導
  - HSTS設定
  - TLS 版本

其他建議事項：

- 定期監控與更新
- 安全漏洞掃描
- 安全日誌紀錄

1.5. 網站設置錯誤樣態

網站設置錯誤樣態

為避免各機關設置HTTPS，或轉導錯誤，列舉以下錯誤態樣：

(1)未關閉HTTP，且未設定轉導。

(2)設置數秒後才轉導至HTTPS。

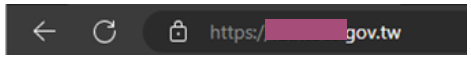
(3)僅首頁設定轉導。

(4)網頁內有HTTP元素。

(5)憑證串鍊中斷。

(1)未關閉HTTP，且未設定轉導，

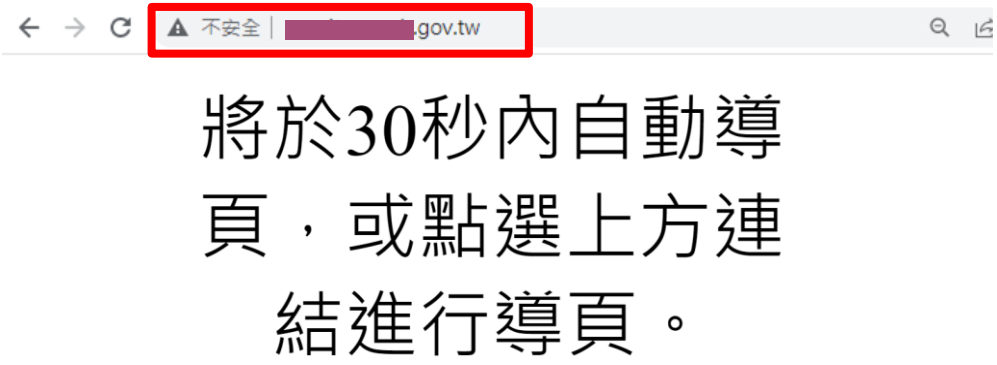
1. 以某網站為例，同時開放HTTP/HTTPS連線。
   - HTTP 連線 

   - HTTPS連線 

2. 建議關閉HTTP，或設定轉導至HTTPS。

(2)設置數秒後才轉導至HTTPS

1. 以某平台為例，進入HTTP頁面後，數秒才自動導轉到HTTPS。
2. 請改為立即跳轉，以保持連線安全性，或關閉HTTP。



(3)僅首頁設定轉導

1. 僅首頁設置轉導，但網站分頁未設定，如從其他分頁進入網站，將不會自動轉跳至HTTPS安全連線。
2. 請依據建議於站台設定301/302轉導、HSTS ，或關閉HTTP連線。

(4)網頁內有HTTP元素

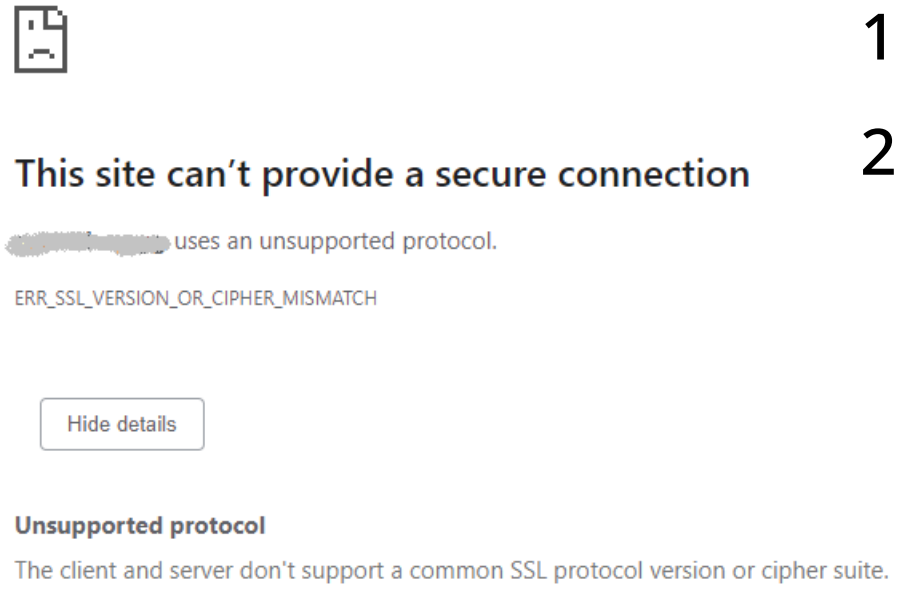
1. 雖已使用HTTPS安全連線，但因網頁內容元素(例如圖檔、影片來源為HTTP站台)，部分瀏覽器會顯示不安全。

   例:
2. 請檢查網頁元素，移除所有HTTP元素，改為HTTPS。

(5)憑證串鍊中斷

1. 如憑證串鍊中斷，瀏覽器將出現不信任警告。
2. 請依伺服器軟體憑證安裝手冊，重新設定憑證串鍊。

(6)過時的連線演算法



1. 瀏覽器警告(SSL/TLS錯誤)
2. 加密通訊協定失敗

1.6. 其他建議

其他建議

1. 請貴機關評估網站關閉**HTTP**之可行性，以提升網站之安全性。
2. 請貴機關清查是否有不再使用的網站(例:為舉辦活動建立之網站)，並將該網站下架且至**GSN**註銷域名，以避免作為駭客攻擊的跳板。
3. 因政府機關TLS憑證效期為1年，請記得於逾期前申請新的憑證，也可使用本部ACME工具自動換發新的憑證。

2. GTLSCA ACME

GTLSCA ACME

2.1. 協定說明

2.2. 目的

2.3. 使用測試環境

2.4. 設定流程

2.5. 憑證更新

2.6. 變更**Certbot**註冊**Email**

2.7. 常見錯誤處理

2.8. FAQ

2.1. 協定說明

協定說明

1. 自動憑證管理環境 (Automatic Certificate Management Environment, ACME)是一種用於自動化獲取和管理TLS憑證的協議，達成自動更新TLS憑證之功效並降低忘記於憑證到期前申請之風險。
2. ACME引入自動化機制，允許網站管理員使用ACME客戶端工具來自動獲取、驗證和部署憑證。
3. Certbot，即是實作ACME客戶端的工具之一。

協定說明

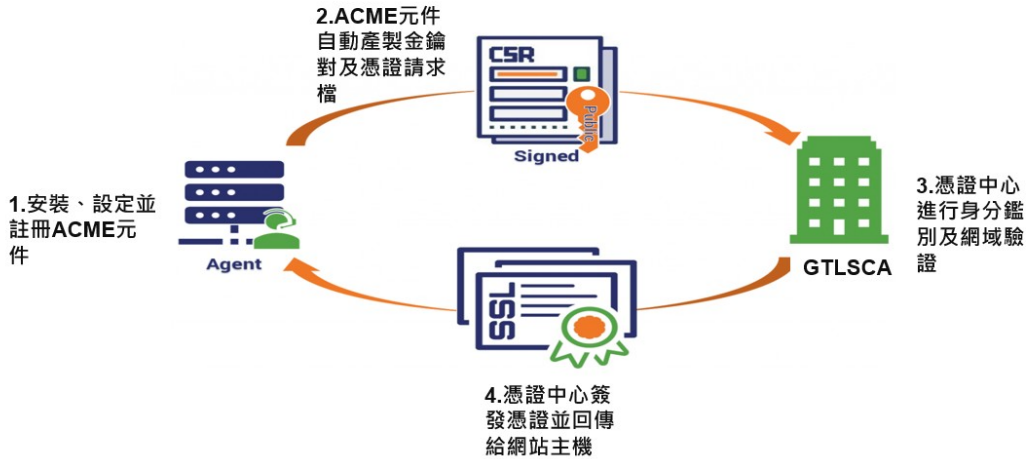
4. 因ACME機制為非必要性之功能，且需額外安裝用戶端元件，因此機關單位應自行評估導入之必要性，建議考量之方向如下：

- 機關單位管理的網站若可用性之要求較高，可考量導入ACME自動換發，透過自動換發，可以排除因人為管理疏忽，導致網站憑證過期造成網站無法連線瀏覽之情況。
- 若網站對於機密性及完整性之要求較高，安裝第三方元件有可能增加額外資安風險，故建議每年手動申請換發，以降低因安裝第三方元件產生之風險。
- 若管理之網站憑證數量較多，且使用單網域憑證，可考量使用ACME自動換發，可有效降低憑證申請及之行政成本以及憑證生命週期管理之複雜度。

2.2. 目的

目的

如何使用本部**ACME**客戶端工具**(Certbot)**，在不同網站伺服器的憑證申請操作，且與**GTLSCA**進行用戶介接及認證。



2.3. 使用測試環境

使用測試環境 **(1/2)**

- 作業系統（**CentOS**、**Ubuntu**、**Windows Server 2016**）
- 網站伺服器 **(Apache**、**Nginx**、**Tomcat**、**IIS)**
- **Certbot (1.11.0** 以上**)**

使用測試環境 **(2/2)**

- Apache
  - Case 1 : Apache x CentOS 7
  - Case 2 : Apache x CentOS 8
  - Case 3 : Apache x CentOS Stream 9
  - Case 4 : Apache x Ubuntu
- Nginx
  - Case 5 : Nginx x CentOS 7
  - Case 6 : Nginx x CentOS 8
  - Case 7 : Nginx x CentOS Stream 9
  - Case 8 : Nginx x Ubuntu
- IIS
  - Case 9 : IIS x Windows Server 2016
- Others
  - Case 10 : Tomcat & 其他不支援憑證安裝網站伺服器 x Linux系統

2.4. 設定流程

操作說明(1/3)

(一) 安裝ACME用戶端工具(本部提供之Certbot)

(二) 利用Certbot 申請帳號

(三) 預註冊GTLSCA ACME用戶帳號。

用戶須填寫各項預註冊資料，包含：欲使用之Certbot用戶端帳號、網域及各項聯絡人資料。再透過GCA卡片簽章，後台系統通過審驗後，認證為合法用戶。

操作說明**(2/3)**

(四) 使用**Certbot** 向**GTLSCA ACME**申請憑證

1.透過**Certbot**，連接至**GTLSCA ACME**伺服器，註冊欲申請網域名稱驗證。

2.網域憑證所有權驗證，通過後進行憑證申請。

3.憑證安裝與更新。

- 自動更新憑證機制支援**Apache**、**Nginx**。

GTLSCA ACME
2.4. 設定流程

2.4.1. 安裝**Certbot**

2.4.2. Certbot 帳號申請

2.4.3. 預註冊**GTLSCA ACME**用戶帳號

2.4.4. 憑證申請

2.4.1. 安裝Certbot

安裝**Certbot (1/20)**

- 用戶端伺服器皆須開啟對外連線**port：443**
- **Certbot執行時皆須以root權限執行。**
- 檢查安裝版本

  指令：**certbot --version**

```
[osboxes@osboxes dist]$ certbot --version
certbot 2.3.0
```

(實際顯示的版本，視您所安裝版本而定**)**

安裝**Certbot (2/20)**

Case 1 : Apache x CentOS 7

- 指令： yum install python-certbot-apache
- 安裝Certbot for apache。
- 模組mod_ssl 可一併安裝，避免安裝憑證時發生錯誤。

安裝**Certbot (3/20)**

Case 2 : Apache x CentOS 8

- 指令： dnf install python3-certbot-apache
- 安裝Certbot for apache。
- 模組mod_ssl 可一併安裝，避免安裝憑證時發生錯誤。
- 在CentOS 8環境需使用python 3.6以上。

安裝**Certbot (4/20)**

Case 3 : Apache x CentOS Stream 9

- 指令： yum install python-certbot-apache
- 可仿效CentOS 8使用yum或dnf進行安裝。
- 模組mod_ssl 可一併安裝，避免安裝憑證時發生錯誤。

安裝Certbot (5/20)

Case 3 : Apache x CentOS Stream 9

- 若使用自編譯的rpm檔(版本:2.3.0)進行安裝，執行環境需Python3.7以上。
- 自編譯的rpm檔需安裝：
  (1) 指令： rpm -ivh certbot-2.3.0-1.noarch.rpm
  (2) 指令： rpm -ivh acme-2.3.0-1.noarch.rpm
  (3) 指令： rpm -ivh certbot-apache-2.3.0-1.noarch.rpm

安裝**Certbot (6/20)**

Case 3 : Apache x CentOS Stream 9

```
[osboxes@osboxes dist]$ sudo rpm -ivh certbot-2.3.0-1.noarch.rpm
[sudo] password for osboxes:
error: Failed dependencies:
        ConfigArgParse>=0.9.3 is needed by certbot-2.3.0-1.noarch
        acme>={version} is needed by certbot-2.3.0-1.noarch
        configobj>=5.0.6 is needed by certbot-2.3.0-1.noarch
        cryptography>=2.5.0 is needed by certbot-2.3.0-1.noarch
        distro>=1.0.1 is needed by certbot-2.3.0-1.noarch
        josepy>=1.13.0 is needed by certbot-2.3.0-1.noarch
        parsedatetime>=2.4 is needed by certbot-2.3.0-1.noarch
        pyrfc3339 is needed by certbot-2.3.0-1.noarch
        pytz>=2019.3 is needed by certbot-2.3.0-1.noarch
```

安裝時有缺乏相依的**python**模組導致無法順利安裝

安裝**Certbot (7/20)**

Case 3 : Apache x CentOS Stream 9

缺乏相依的python模組時 :

- 需先安裝相依的模組，或在指令後加上 **--nodeps --force**先行安裝 (但相依python模組仍是需要安裝)。
- 缺乏的元件可使用pip方式進行安裝, 例如 :

  pip install ConfigArgParse

安裝Certbot (8/20)

Case 4 : Apache x Ubuntu

- Debian體系下(以Ubuntu 22.04為例)，若使用自編譯的deb檔(版本:2.3.0)進行安裝，執行環境需Python3.7以上。
- 自編譯的deb檔需安裝：
  (1) 指令： apt install python3-acme_2.3.0-1_all.deb
  (2) 指令： apt install python3-certbot_2.3.0-1_all.deb
  (3) 指令： apt install python3-certbot-apache_2.3.0-1_all.deb

安裝Certbot (9/20)

Case 5 : Nginx x CentOS 7

- 指令： yum install python-certbot-nginx
- 可仿效CentOS 8使用yum或dnf進行安裝。

安裝**Certbot (10/20)**

Case 6 : Nginx x CentOS 8

- 指令： dnf install python3-certbot-nginx
- 需在CentOS 8環境需使用python 3.6以上。

安裝**Certbot (11/20)**

Case 7 : Nginx x CentOS 9 Stream

- 指令： yum install python-certbot-nginx
- 可仿效CentOS 8使用yum或dnf進行安裝。

安裝Certbot (12/20)

Case 8 : Nginx x Ubuntu

- Debian體系下(以Ubuntu 22.04為例)，若使用自編譯的deb檔(版本:2.3.0)進行安裝，執行環境需Python3.7以上。
- 自編譯的deb檔需安裝：
  (1) 指令： apt install python3-acme_2.3.0-1_all.deb
  (2) 指令： apt install python3-certbot_2.3.0-1_all.deb
  (3) 指令： apt install python3-certbot-nginx_2.3.0-1_all.deb

安裝**Certbot (13/20)**

Case 9 : IIS x Windows Server 2016

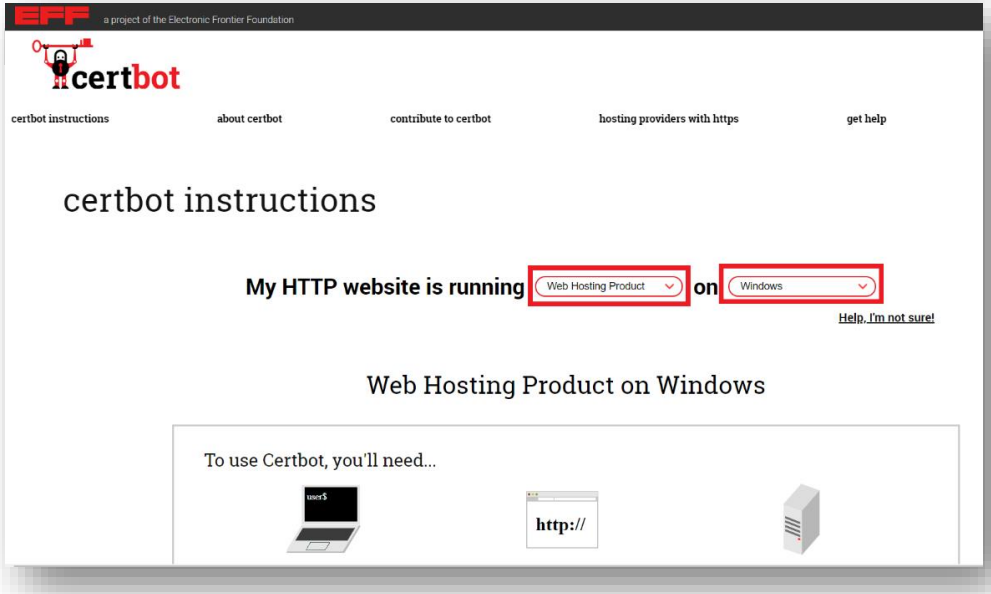
- Certbot於IIS不支援憑證更新。
- 有以下有兩個安裝方式：

  (1) 自編譯安裝檔：certbot-beta-installer-win_amd64.exe

  (2) certbot官方網站下載安裝檔

安裝Certbot (14/20)

Case 9 : IIS x Windows Server 2016

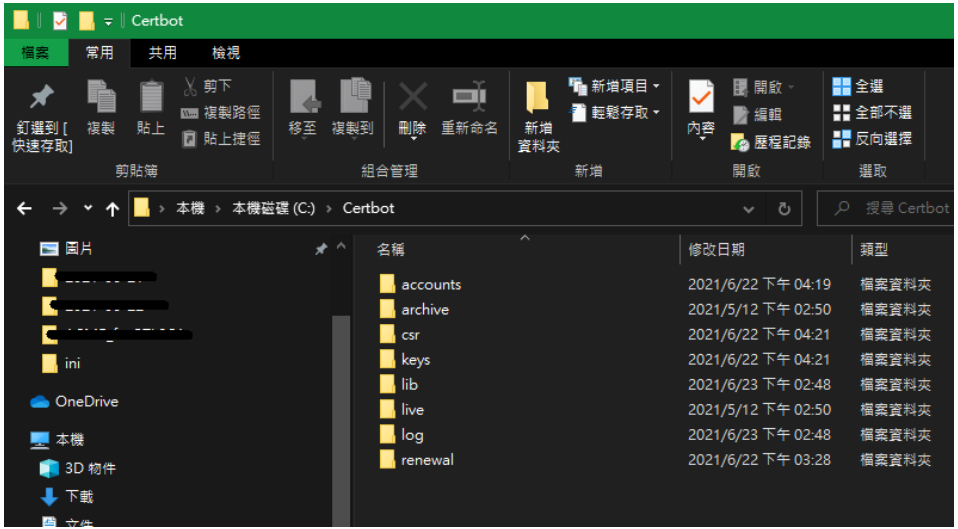


選擇Web Hosting Product及安裝在windows作業系統，下載後直接安裝。

https://certbot.eff.org/

安裝Certbot (15/20)

Case 9 : IIS x Windows Server 2016



安裝後，在C:\Certbot會存放ACME帳號及其他憑證資訊。

安裝**Certbot (16/20)**

Case 10 : Tomcat或其他不支援憑證安裝之網站伺服器 x Linux

- 指令 : yum install certbot
- 因Tomcat不支援憑證安裝，因此不指定網站伺服器。其他不支援憑證安裝的網站伺服器亦可使用此方法。

安裝**Certbot (17/20)**

Case 10 : Tomcat或其他不支援憑證安裝之網站伺服器 x Linux

- 若使用自編譯的rpm檔(版本:2.3.0)進行安裝，執行環境需Python3.7以上。
- 自編譯的rpm/deb檔需安裝：

➢ RedHat/CentOS 安裝指令：

(1) rpm -ivh certbot-2.3.0-1.noarch.rpm

(2) rpm -ivh acme-2.3.0-1.noarch.rpm

➢ Debian/Ubuntu安裝指令：

(1)apt install python3-acme_2.3.0-1_all.deb

(2)apt install python3-certbot_2.3.0-1_all.deb

安裝**Certbot (18/20)**

Case 10 : Tomcat或其他不支援憑證安裝之網站伺服器 x Linux

```
[osboxes@osboxes dist]$ sudo rpm -ivh certbot-2.3.0-1.noarch.rpm
[sudo] password for osboxes:
error: Failed dependencies:
        ConfigArgParse>=0.9.3 is needed by certbot-2.3.0-1.noarch
        acme>={version} is needed by certbot-2.3.0-1.noarch
        configobj>=5.0.6 is needed by certbot-2.3.0-1.noarch
        cryptography>=2.5.0 is needed by certbot-2.3.0-1.noarch
        distro>=1.0.1 is needed by certbot-2.3.0-1.noarch
        josepy>=1.13.0 is needed by certbot-2.3.0-1.noarch
        parsedatetime>=2.4 is needed by certbot-2.3.0-1.noarch
        pyrfc3339 is needed by certbot-2.3.0-1.noarch
        pytz>=2019.3 is needed by certbot-2.3.0-1.noarch
```

安裝時有缺乏相依的**python**模組導致無法順利安裝

安裝**Certbot (19/20)**

Case 10 : Tomcat或其他不支援憑證安裝之網站伺服器 x Linux

缺乏相依的**python**模組時 :

- 需先安裝相依的模組，或在指令後加上 **--nodeps --force**先行安裝 (但相依**python**模組仍是需要安裝)。

安裝Certbot (20/20)

Certbot安裝目錄與資料夾說明

- 使用CentOS/Ubuntu : 目錄於/etc/letsencrypt ; 使用Windows : 目錄於C:\Certbot 。
- accounts：帳號相關資訊。針對同一ACME Server僅有一個。
- csr ：憑證申請檔。
- live：目前的憑證、憑證串鍊、私鑰。
- renewal : 憑證更新相關設定。
- archive : 憑證(包含舊的憑證) 。
- keys : 私有金鑰(包含未申請成功，若未成功下載憑證，可自行下載憑證，並從此資料夾找到私鑰) 。

2.4.2. Certbot 帳號申請

Certbot 帳號申請(1/2)

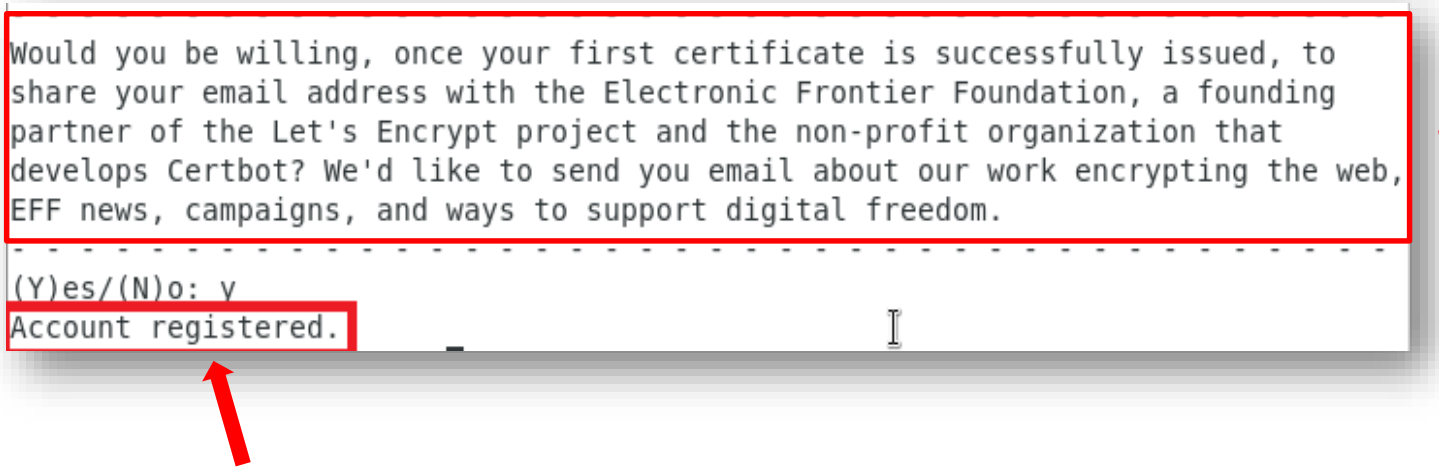
申請指令

- certbot register --agree-tos -m 系統管理員電子郵件信箱 --server https://gtlscaweb.nat.gov.tw/ACMEWeb/directory

指令參數說明

- --agree-tos：同意服務條款。(e.g.使用Certbot進行帳號申請、系統管理員需開啟80或443 port進行網域驗證作業、須使用 GCA卡片進行帳號預註冊...)
- -m：系統管理員電子郵件信箱。若申請ACME帳號成功，會寄送帳號到此電子郵件信箱。
- --server：指向GTLSCA ACME 伺服器。

Certbot 帳號申請(2/2)



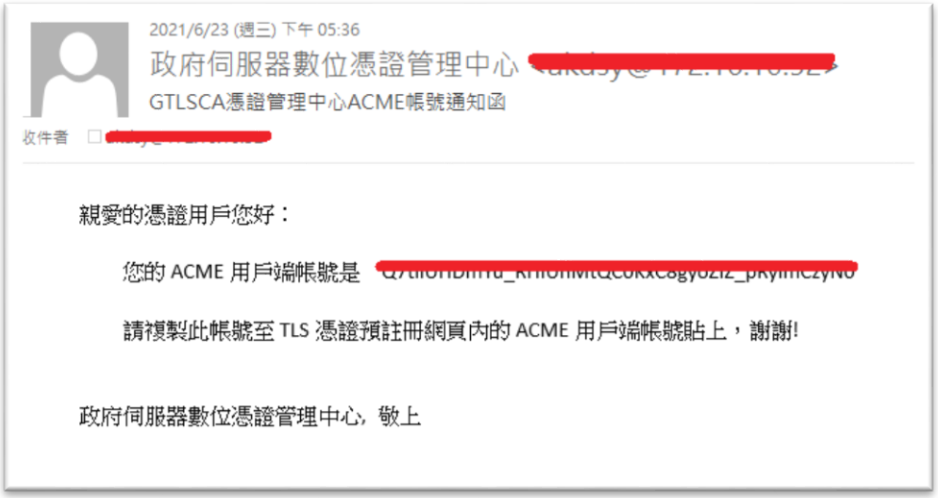
- 是否支援數位自由，請自行決定(建議不要)。

- 如註冊成功，可看見此訊息。

2.4.3. 預註冊GTLSCA ACME用戶帳號

預註冊 **(1/7)**

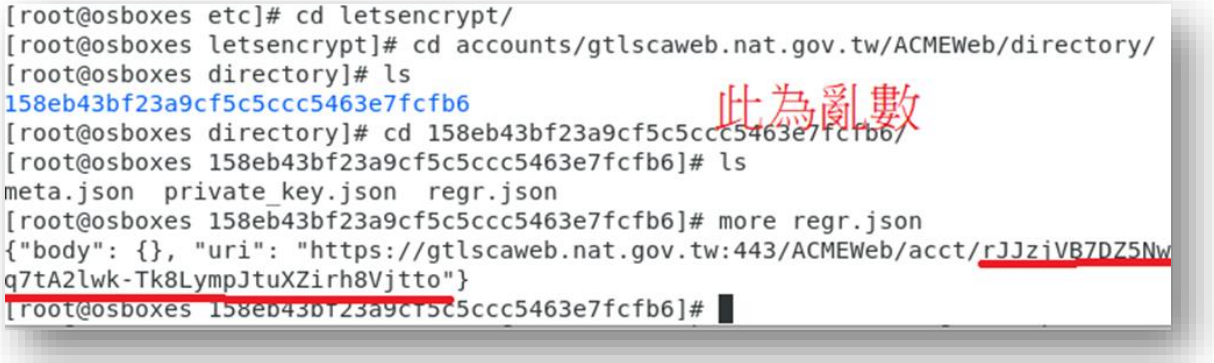
Step 1：取得ACME之註冊帳號



- 帳號申請成功之後，會在您申請帳號時填入的email中收到ACME帳號通知函，在此步驟使用GCA卡片綁定GTLSCA憑證申請所需資料。

預註冊 **(2/7)**

沒有收到帳號通知信之解決方式 - Linux



- 至 **/etc/letsencrypt/accounts/gtlscaweb.nat.gov.tw/ACMEWeb/directory/亂數/regr.json**，開啟檔案取得**ACME帳號**。

預註冊 **(3/7)**

沒有收到帳號通知信之解決方式 - Windows

- 至 **C:\Certbot\accounts\gtlscaweb.nat.gov.tw\ACMEWeb\directory\亂數\regr.json**，開啟檔案取得**ACME**帳號。

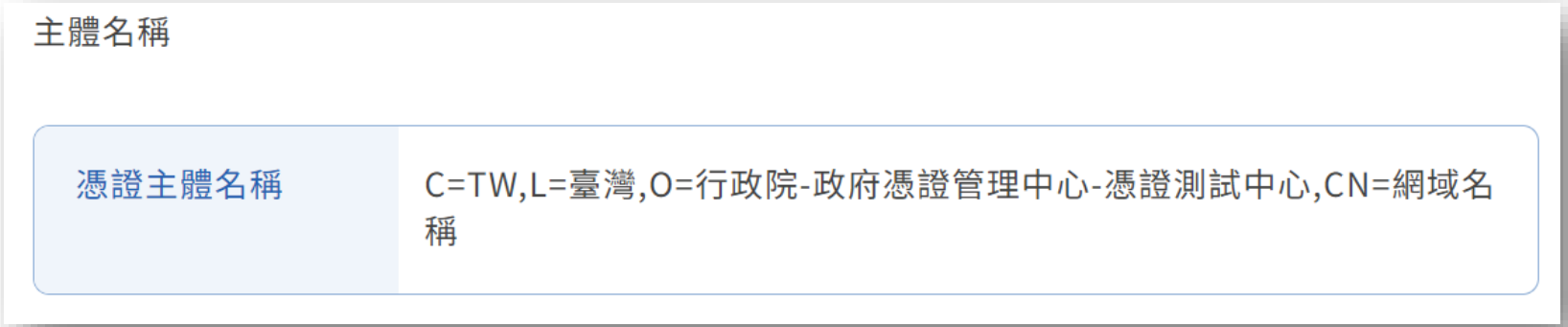
預註冊 **(4/7)**

Step 2：至GTLSCA預註冊網頁，使用GCA卡片進行預註冊程序

- GTLSCA預註冊網址：https://gtlscaweb.nat.gov.tw/GCP/views/ACME/ACME_apply.html

預註冊 **(5/7)**

Step 3：確認主體名稱是否有誤



- 原OU欄位劃歸至O欄位下，不再額外描述。

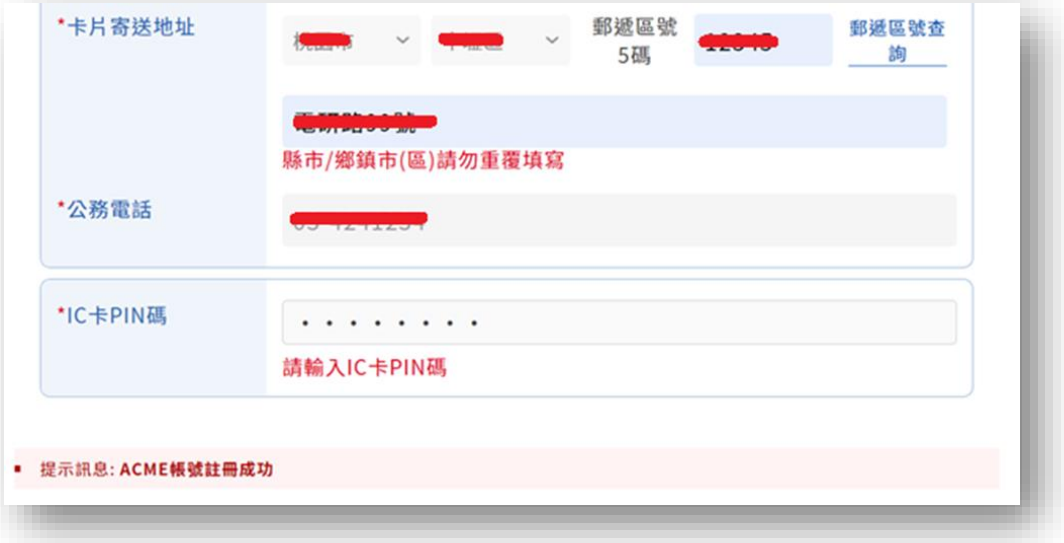
預註冊 **(6/7)**

Step 4 : 填寫各項必要資料

- 填寫網站名稱、**ACME**帳號及憑證聯絡人資料。
- 一個用戶帳號，可以用來申請不同的網站名稱。

預註冊 **(7/7)**

Step 5：上傳預註冊資料



- 可在提示訊息，檢視是否預註冊成功。

2.4.4. 憑證申請

憑證申請 **(1/11)**

示範說明

- **Certbot**目前只支援**Apache**及**Nginx**的自動憑證安裝。以下示範使用**certbot -i apache**、**certbot -i nginx**自動憑證安裝。
- 針對**IIS**、**Tomcat**、**Wildfly**等網站伺服器，**Certbot**不支援自動憑證安裝，但支援透過指令下載憑證，因此，須請網站管理員手動進行憑證安裝作業。

憑證申請 (2/11)

certbot 指令參數說明

- certonly : 僅下載憑證至certbot定義之特定位置(憑證申請成功才會提示)，不進行憑證安裝。
- -i : 指定要進行憑證安裝的網站伺服器，目前此項參數僅支援 apache或nginx。
- --webroot :
  - 使用既有網站伺服器進行網域驗證。
  - Certbot會將網域驗證資料放置於此根目錄下並建立資料夾路徑 /.well-known/acme-challenge/ 。網站管理員須開啟http/https連入權限，讓ACME伺服器可以進行網域所有權驗證作業(驗證後certbot會刪除此路徑下的網域驗證資料)。
  - GTLSCA 的ACME 伺服器支援驗證網域所有權於port 80或443。

憑證申請 **(3/11)**

certbot 指令參數說明

- **--standalone :** 若網站伺服器尚未完備，可使用**certbot**建構網站伺服器進行網域驗證。使用此參數則需占用**port 80**，與**--webroot**使用方式不同。
- **--preferred-challenge :** 網域所有權驗證方式，目前僅支援**http-01**。
- **update_account :** 變更**email**帳號。
- **-d :** 欲申請**TLS**憑證的網域。
- **--key-type :** 金鑰類別 **rsa**或**ecdsa**，目前**GTLSCA**僅支援**rsa** 。
- **--rsa-key-size :** 金鑰類別 **rsa**的金鑰長度。

憑證申請 **(4/11)**

自動憑證申請及安裝 - **Apache**

- 指令：

  certbot -i apache --webroot –w 網站根目錄 --key-type rsa --rsa-key-size 2048 --preferred-challenge 網域驗證方式 -d 申請網域 --server https://gtlscaweb.nat.gov.tw/ACMEWeb/directory

```
[osboxes@osboxes Downloads]$ sudo certbot -i apache --webroot -w /var/www/html/ --key-type rsa --rsa-key-size 2048 --preferred-challenge http-01 -d [redacted] --server https://gtlscaweb.nat.gov.tw/ACMEWeb/directory
```

憑證申請 **(5/11)**

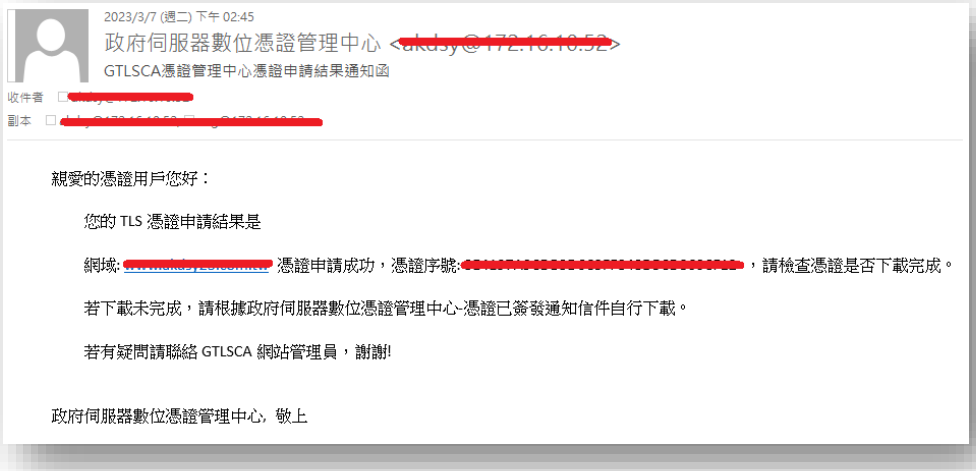
自動憑證申請及安裝 - **Apache**

```
[osboxes@osboxes Downloads]$ sudo certbot -i apache --webroot -w /var/www/html/ --key-type rsa --rsa-key-size 2048 --preferred-challenge http-01 -d [redacted] -
-server https://gtlscaweb.nat.gov.tw/ACMEWeb/directory
Saving debug log to /var/log/letsencrypt/letsencrypt.log

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/[redacted]/fullchain.pem
Key is saved at:         /etc/letsencrypt/live/[redacted]/privkey.pem
This certificate expires on 2024-03-07.
These files will be updated when the certificate renews.

Deploying certificate
Successfully deployed certificate for [redacted] to /etc/httpd/conf.d/ssl.conf
Your existing certificate has been successfully renewed, and the new certificate has been installed.

NEXT STEPS:
- The certificate will need to be renewed before it expires. Certbot can automatically renew the certificate in the background, but you may need to take steps to enable
 that functionality. See https://certbot.org/renewal-setup for instructions.

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
If you like Certbot, please consider supporting our work by:
 * Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
 * Donating to EFF:                    https://eff.org/donate-le
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

- 憑證申請成功，檔案會下載至**/etc/letsencrypt/live/** 資料夾下，包含**TLS**憑證及憑證串鍊。

憑證申請 **(6/11)**

自動憑證申請及安裝 - Apache



- 憑證申請成功或失敗皆會寄發通知信，請網站管理者確認網站狀況。

憑證申請 **(7/11)**

自動憑證申請及安裝 - Nginx

- 指令：

  certbot -i nginx --webroot –w 網站根目錄 --key-type rsa --rsa-key-size 2048 --preferred-challenge 網域驗證方式 -d 申請網域 --server https://gtlscaweb.nat.gov.tw/ACMEWeb/directory

```
[osboxes@osboxes ~]$ sudo certbot -i nginx --webroot -w /usr/share/nginx/html/ --key-type rsa --rsa-key-size
2048 --preferred-challenges http-01 -d [redacted] --server https://gtlscaweb.nat.gov.tw/ACMEWeb/direct
ory
```

憑證申請 **(8/11)**

自動憑證申請及安裝 - Nginx

- 憑證申請成功，檔案會下載至**/etc/letsencrypt/live/** 資料夾下，包含**TLS**憑證及憑證串鍊。**(**畫面同**Apache**執行結果**)**
- 憑證申請成功或失敗皆會寄發通知信，請網站管理者確認網站狀況。**(**畫面同**Apache**執行結果**)**

憑證申請 **(9/11)**

僅憑證申請 - 不支援憑證自動安裝的網站伺服器 (IIS, Tomcat, Wildfly, etc.)

- 指令：

  certbot certonly --webroot -w 網站根目錄 --key-type rsa --rsa-key-size 2048 --preferred-challenge http-01 -d 申請網域 --server https://gtlscaweb.nat.gov.tw/ACMEWeb/directory

- 僅申請下載憑證，憑證安裝作業須請網站管理員手動進行。

憑證申請 **(10/11)**

僅憑證申請 - 不支援憑證自動安裝的網站伺服器 (IIS, Tomcat, Wildfly, etc.)

```
[osboxes@osboxes ROOT]$ sudo certbot certonly --webroot -w /var/lib/tomcat/webapps/ROOT --key-type rsa --rsa-key-size 2048 --preferred-challenge http-01
-d [redacted] --server https://gtlscaweb.nat.gov.tw/ACMEWeb/directory
[sudo] password for osboxes:
Saving debug log to /var/log/letsencrypt/letsencrypt.log

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/[redacted]/fullchain.pem
Key is saved at:         /etc/letsencrypt/live/[redacted]/privkey.pem
This certificate expires on 2024-03-08.
These files will be updated when the certificate renews.

NEXT STEPS:
- The certificate will need to be renewed before it expires. Certbot can automatically renew the certificate in the background, but you may need to take
steps to enable that functionality. See https://certbot.org/renewal-setup for instructions.

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
If you like Certbot, please consider supporting our work by:
 * Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
 * Donating to EFF:                    https://eff.org/donate-le
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

- 以使用**Tomcat**為例，當憑證申請成功，檔案會下載至**/etc/letsencrypt/live/**資料夾下，包含**TLS**憑證及憑證串鍊。

憑證申請 **(11/11)**

僅憑證申請 - 不支援憑證自動安裝的網站伺服器 (IIS, Tomcat, Wildfly, etc.)

```
IMPORTANT NOTES:
 [0m - Congratulations! Your certificate and chain have been saved at:
   C:\Certbot\live\[redacted]\fullchain.pem
   Your key file has been saved at:
   C:\Certbot\live\[redacted]\privkey.pem
   Your certificate will expire on 2022-06-24. To obtain a new or
   tweaked version of this certificate in the future, simply run
   certbot again. To non-interactively renew *all* of your
   certificates, run "certbot renew"
 - If you like Certbot, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
   Donating to EFF:                    https://eff.org/donate-le
```

- 以使用**IIS**為例，當憑證申請成功，檔案會下載至**C:\Certbot\live** 資料夾下，包含**TLS**憑證及憑證串鍊。

2.5. 憑證更新

憑證更新

說明

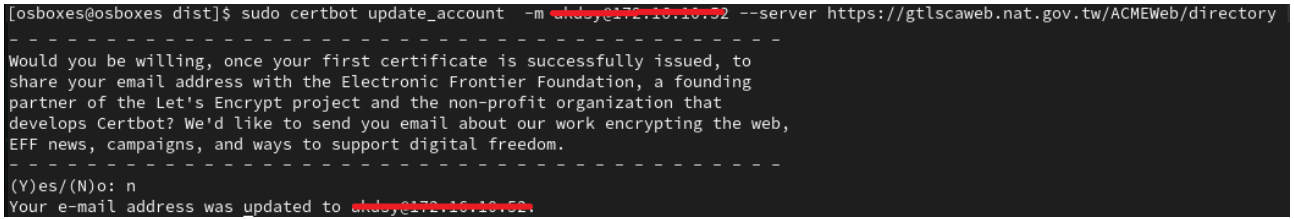
- 指令 : **certbot renew**。
- **certbot**憑證更新前會檢查效期，若非到期**前30天**則不可進行憑證更新作業。
- 可自行設定排程，讓憑證到期前自動更新。
  - **Windows:** 設定工作排程器與批次檔。
  - **Linux:** 建立**shellScript**，並設定到**crontab**排程執行。

2.6. 變更Certbot註冊Email

變更**Certbot**註冊**Email**

說明

- 指令：**certbot update_account -m** 用戶系統管理員電子郵件信箱 **--server https://gtlscaweb.nat.gov.tw/ACMEWeb/directory**
- 執行時仍會詢問您是否支援數位自由，並不影響後續**Email**變更作業。完成後會再次寄發**ACME**帳號信件。



2.7. 常見錯誤處理

常見錯誤處理

常見錯誤訊息與處理方式

- **accountDoesNotExist :** 帳號不存在，請重新申請帳號。
- **userEmailRequired :** 管理員電子郵件信箱僅能輸入一個。
- **userActionRequired :** 須同意用戶條款。
- **unauthorized :** 帳號狀態錯誤。若狀態顯示**pending**，請先進行預註冊。
- **Incorrect response :** 網域所有權驗證失敗。請確認執行**Certbot**之主機是否擁有此網域，且是否可以連入「**http(s)://網域/.well-known/acme-challenge/**驗證資料」。且須注意是否因**load-balance**設定而連線到非**Certbot**執行之主機進行驗證。

2.8. FAQ

FAQ (1/5)

Q1：安裝了Certbot，一台伺服器可以申請多個ACME帳號嗎? (每次註冊使用不同信箱註冊)

A1：Certbot對於帳號的認定是CA，同一個CA只能有一個帳號。以本服務的情境，只有一個CA (GTLSCA)，故只有一組ACME帳號。

FAQ (2/5)

Q2：變更Certbot註冊之Email，重新寄發的ACME帳號是會跟之前一樣嗎？

A2：一樣。更換Certbot註冊的Email，只是更換憑證簽發結果通知信的窗口，不影響原ACME帳號。

FAQ (3/5)

Q3：預註冊成功後，憑證是立即簽發？還是多久可以收到簽發之憑證序號？

A3：

FAQ (4/5)

Q4：預註冊可以多網域註冊申請?

A4：可以，但多網域須集中在certbot執行的主機上，網域驗證才不會有問題。但目前僅限開放單網域憑證申請。

FAQ (5/5)

Q5：若以ACME自動換發憑證，之後可以再使用插卡或發文之方式申請TLS憑證嗎？

A5：可以，兩種申請方式彼此不互相干擾，憑證取決於最後一次執行憑證的設定方式而定， 例如以下使用情境：

【發文/插卡申請】→【ACME自動換發】→【發文/插卡申請】

則最終使用的憑證，是來自於【發文/插卡申請】 。

THANKS FOR YOUR ATTENTION

moda

數位發展部

Ministry of Digital Affairs