Nginx HTTP Server SSL 服務運轉 OCSP Stapling 之設定與測試

本說明書適用於 Nginx 環境下之 SSL 伺服器軟體憑證安裝，並假設 Nginx HTTP Server 係執行於 Linux。本說明書的安裝程序，已經在 Nginx-1.7.4 版測試過，您所使用的版本或環境可能與本版本有所差異，若是如此則請參考您的 Web Server 及 SSL 模組相關 使用手冊，適度調整 SSL 伺服軟體憑證安裝步驟。Nginx Server 官方網站請參考 http://nginx.org/

1. 本次設定與測試環境如下說明
   - Nginx HTTP Server 架設在 CentOS 6.3 64 位元作業系統上
   - 整體網站服務之運轉環境為 Nginx HTTP Server 1.7.4 + OpenSSL 1.0.1i + PHP 5.3.29

2. Nginx 系統環境之設定說明
   - Nginx HTTP Server 必須先設定好 SSL 服務
   - 修改 Nginx HTTP Server 組態設定檔案 conf/nginx.conf，加入 OCSP Stapling Configuration，如下設定

```
server {
    listen          443 ssl;
    server_name    192.168.133.250;
    ssl_certificate          /export/nginx-1.7.4/conf/Certs/SSL250.pem;
    #nginx 本身既有的 SSL Base64 憑證，
    ssl_certificate_key       /export/nginx-1.7.4/conf/Certs/SSL250.key;
    #nginx 本身既有的 SSL Base64 憑證之相對應的私鑰

    root     /export/www/htdocsSSL/;     #SSL 主要服務網站目錄
    index    index.html index.htm index.php;   #網站入口檔案

    #Nginx OCSP Stapling 設定段落
    ssl_stapling on;
    ssl_stapling_verify on;
    ssl_trusted_certificate /export/nginx-1.7.4/conf/TrustCertChain.pem;

    #此 TrustCertChain.pem 包含內容有 SSL 憑證簽發的 CA Base64 憑證
```

與 RootCA Base64 憑證，放置有 SSL 憑證的可信憑證串列

}

- TrustCertChain.pem　Example: 實際上就是 GRCA CA 憑證與 GCA 憑證，或者 eCA CA 憑證與 Public CA 憑證

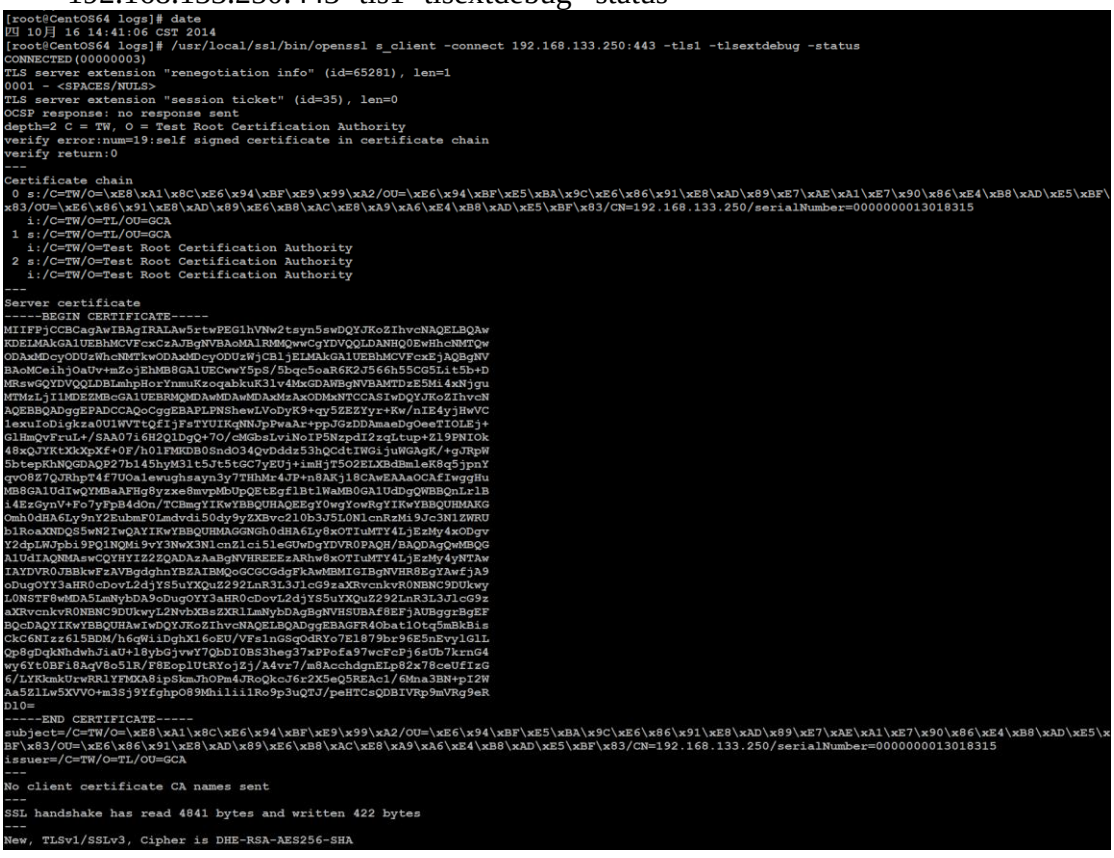
-----BEGIN CERTIFICATE-----
MIIFPzCCAyegAwIBAgIRALm9ZUDUqkyTYRtm36nZyNEwDQYJKoZIhvcNAQELBQAw
OTELMAkGA1UEBhMCVFcxKjAoBgNVBAoMIVRlc3QgUm9vdCBDZXJ0aWZpY2F0aW9u
IEF1dGhvcml0eTAeFw0xNDA3MzAwMTM2NTNaFw0zNDA3MzAwMTM2NTNaMDkxCzAJ
BgNVBAYTAlRXMSowKAYDVQQKDCFUZXN0IFJvb3QgQ2VydGlmaWNhdGlvbiBBdXRo
b3JpdHkwggIiMA0GCSqGSIb3DQEBAQUAA4ICDwAwggIKAoICAQCHwy43pDmpGBZw
bh2T1g/qL4CBlnMDX0jgh37z7LUNQUr4DrP4Sim3nZe/fClGLmxZNDAAVnGqmMkJ
+MvAL+gfAxONfVxG3GFinkp10RdIUQr7Hxlb07w9W70cD+2q/kwKdvVdQgZqxs3X
eOFWwc+vPLevpzFn6zk+bXH7UYNnZOGc034tG7AgFuyUBuZYBCLpwxS/eBWrG4b0
MHd52tqJnamfUnX19zw7s/bGD7Nto2pQTowEy+5w3mAuw1hxwaeVlbBc/9QxHQ9P
oMx9sFKapA1RsWb+jWONEAIS5FO23uRFSAc2Wqxd6Zt7zZdxYR+CrzsIm3MwzaiW
HDKYXwDhJCF9ZaMhxVSULZVpSLCmlBGmk4SDQG80l5NWjIhTVHECEuV5Rkwv7tap
uDCe/AApZdc9+Qm4ggnW8b+IXh84JLbQqkjV51I2A/YXEOat0Joe0iSF33UiDm9b
3SYXCpx3LAMIxEbD9RevxgguiNwan/F9B67LctVCabkjP9P5xnsG0LZ763svyUXY
P2f29QFDmkwHlKFsCfZGfUZeeCoVihtp3E4q0ZUwt5V5UMgfTWHcKwkEyzZEarvJ
wfP8NrLVHFanI0fXTrLfJGECZgBeEiVWC+TafODyRRYZfKRhsULkFvZI5nOuD/+l
MgVmM1HlWVu6BoJ6hRe5abyxyowiLQIDAQABo0IwQDAPBgNVHRMBAf8EBTADAQH/
MB0GA1UdDgQWBBTbCcK+fOl7U5Ee4WGt/12xOr8+YDAOBgNVHQ8BAf8EBAMCAQYw
DQYJKoZIhvcNAQELBQADggIBAFu/cXDOYuLGXh6nLlfe7jI0UeZRzzbOCx3/Ryy9
wY3aZfkYQdnqjdXW3SSLfow4AfcbecEaw6RvD2BqW4uMv5x0CpAaQN+FxhfpcW8I
+ud41upDqSno9K0hJApEtNMNqKAeQsmu687BMqt+IVzupoLeSuYYsFmmYhBvpLBw
dTPfykaeoUF5O318+0y4ay1E2cAp5d8MYb+T9Igj70TwZrqpLYWn8ibhDRvA0yMb
znqQQUjiIovip0NVYO/lsW28wiCPvnNhcjFk811sdV6z9ZuLxzJNnJNlp6jYooiP
UW5n8ULFs0fe08/QGsViwA06A4lhh+w3kg2NmIj3HAdZVaY8HGl6sHn3RLzmKV51
Lz7GqeTuzD14OX/CfRYWkEzlYa226Ql6OvLHTgJV3aeeojxfjPTGtdv8lxLWNoOj
zHyGHv+Ch6gH8UFd7N1bygtTc1s3XFqz8ZpCba76bAoG2SAOTrlKwM7bzB/RWVy5
2hwm+6rHrmDzOI7pKdYEAgu9qf25ubvcKAMa4m0yDuhWhHUCDNDJuBd2X+ppiUAV
pfSs8S3B5wtYvWucFe4AyqUCE/xUv3Kw48Z3E/LQs2PnCXYv/eIRJKoeoc+O6Zbo
bunXlAQS6Yi0Kl9dBPkKS/CA6U/wq68fi1uqSX+zIpOkHo57ey5v1DbApY8Mk5mb
uDtQ
-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----
MIIFDDCCAvSgAwIBAgIRAI7qIPjB9kMs8/r+1XB0udAwDQYJKoZIhvcNAQELBQAw
OTELMAkGA1UEBhMCVFcxKjAoBgNVBAoMIVRlc3QgUm9vdCBDZXJ0aWZpY2F0aW9u
IEF1dGhvcml0eTAeFw0xNDA3MzAwMTUwMDZaFw0yOTA3MzAwMTUwMDZaMCgxCzAJ
BgNVBAYTAlRXMQswCQYDVQQKDAJUTDEMMAoGA1UECwwDR0NBMIIBIjANBgkqhkiG
9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxqeNZRjJccB9p6ZxILnw5VFDCNIWfGKKScUh
9XHplczgkUmfpLN8HVC1VfZfuaXBJdus7xoTBCd4RFF0f+bYJhBMTks6am6soYq4
F/08oad6IbgGSIPYx3wwTY3r86h7NAaO6ZW5e5kFng6jPsB9QwIHGOn07laVknmX
jqyvU+P4W6gR25i7uRmRWKjfW5eAM6TGaHDLYgUEDEn2RTxR0V9uDfyA4XCdxdwo
KjviX0uP0w+PqWD5baLb4SPJlW36e0KsQirmuY3SxLan1TnWtVrwal3naZKwjRG7
EDSa1CzPsgXIDsTIEfnWRsClONaxPsvGbNMfrxkbLbOQbkDpkwIDAQABo4IBHjCC
ARowHwYDVR0jBBgwFoAU2wnCvnzpe1ORHuFhrf9dsTq/PmAwHQYDVR0OBBYEFHg8
yzxe8mvpMbUpQEtEgflBtlWaMA4GA1UdDwEB/wQEAwIBBjAUBgNVHSAEDTALMAkG
B2CGdmUAAwMwEgYDVR0TAQH/BAgwBgEB/wIBADBCBgNVHR8EOzA5MDegNaAzhjFo
dHRwOi8vMTAuMTQ0LjEzMy45Mjo3MDAwL3JlcG9zaXRvcnkxL0NSTDIvQ0EuY3Js
MFoGCCsGAQUFBwEBBE4wTDBKBggrBgEFBQcwAoY+aHR0cDovLzEwLjE0NC4xMzMu
OTI6NzAwMC9yZXBvc2l0b3J5MS9DZXJ0cy9Jc3N1ZWRUb1RoaXNDQS5wN2IwDQYJ
KoZIhvcNAQELBQADggIBABJyWotI8+XTqEfu75/Mc+Vs6Gejclmgdx6eo9CxBUuJ
FaznFlCYa7WWb4VbixsUG9BHcCIzWNCXYV2TyIjSr/6vQpgFX4UKL2xeoKHIBbS1
WGzukq+ECmGLJqclWmRRE9HLHMf2tHwCDdXx2lkisJQJTdAY4gSp1WumTgBr1ujG
VKElY5FTcXC6r7EJiCq/AHo2oJWPNFzcA2O4KeUzl/Jzzg1qILDfF2mkk+lLnzlw
QQxEzOuv7VZnZqHSHryWCAaS5K2UcGVXNBArJ8gN5j3fCaLXX5jFFbYCUAY0Caf9
gT/DwOk7taBDUSRJn0eJ4Xa+w37F8zCVgDzDf7KzJvF5YsnCftuNrqMgUVaP/l3h
TEoWySSRrVIE3WDKOV4IUXYURyb3VdSNdD6JVcOgg9Ylp1Aufl5KL0BLU/kDQ01U
tVjPMQlyA1kUTZ9TllG//oA1A7xOCI7zz6yI7Yl5Q5zoqYuLoZLHi91HLPwvvob8e
RRci9sohp8XJDVheml+1vszR5gd44DyVPgJbOG9+hwmo8soUI+maqdIBKhgMV21Q
6Etp0+/QZnXGmIWFgARCAuy6osoUoVINTPEKtR2r/1xDqKo3eFgvef9q9PwR03dt
giCtVqAy6vrVWWOMO6dczQ5vRYYbEmMXXFA7/aphdjZP3R3BZAyXrSXsXwAkHTTF

-----END CERTIFICATE-----

- 修改與設定完成 Nginx HTTP Server 組態設定，必須重新啟動 Nginx 服務才可以載入新的組態設定之內容。

3. 透過 Openssl s_client 測試 Nginx OCSP Stapling 之運轉測試說明
   - 模擬系統環境如下:
   - 模擬 SSL 網站 https://192.168.133.250
   - 模擬 OCSP Server 回傳 SHA256 Response http://192.168.133.188/cgi-bin/OCSP2/ocsp_server.exe
   - 使用 Openssl s_client 工具進行模擬測試，並且開啟 OCSP Server 的 Apache HTTP Server LOG 與 Nginx HTTP Server 兩個主機畫面進行運轉測試與記錄。
   - 模式測試情形如下：
     (1) 先進行 OCSP Server 與 Nginx Server 兩台測試主機的時間調校，必須為設定為時間一致。
     (2) 由 Nginx Server 上執行第一次 HTTPS 連線測試，因為第一次使用 HTTPS，一定會去 OCSP Server 詢問該憑證之狀態，故畫面比較特別，如下所示：
     時間點為 14:41 分,執行指令/usr/local/ssl/bin/openssl s_client -connect 192.168.133.250:443 -tls1 -tlsextdebug –status



```
Server public key is 2048 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
SSL-Session:
    Protocol  : TLSv1
    Cipher    : DHE-RSA-AES256-SHA
    Session-ID: 016AA8067DECC3068A00D2F3642BA599BC52460CA6E95D2DA020BA7511687253
    Session-ID-ctx:
    Master-Key: BB8D90FF857360EAF6006DA097C09C69D989EC85FE2F2EE48550989562178F409420C84EC95E9E497E26C24DAEB5BC1D
    Key-Arg   : None
    PSK identity: None
    PSK identity hint: None
    SRP username: None
    TLS session ticket:
    0000 - ff c3 9b 54 50 5a 0f 26-2b c7 48 b5 6a 7f 21 00   ...TPZ.&+.H.j.!.
    0010 - c8 8e 23 dc 23 00 51 bf-76 0d 99 54 67 bc 5f 54   ..#.#.Q.v..Tg._T
    0020 - 90 3d a8 fc 43 5f 7a 18-fb 79 0b 93 72 d0 d3 69   .=..C_z..y..r..i
    0030 - a7 ed 71 ee c2 eb a9 fe-b1 a4 07 a9 59 20 f9 e0   ..q.........Y ..
    0040 - 09 1f 4b ba f3 76 49 17-e0 dc 25 ce 8d e2 05 f2   ..K..vI...%.....
    0050 - fa c5 8a be 98 52 94 95-36 24 04 31 4b 08 d3 e0   .....R..6$.1K...
    0060 - ea 20 62 5a 05 0a ad 82-5a 61 4d df d8 a2 bb ac   . bZ....ZaM.....
    0070 - 01 e7 01 b2 e4 4a 25 c9-75 5f ca 76 11 bf a8 16   .....J%.u_.v....
    0080 - fb ba 29 f1 68 3c 33 c8-31 4c 0a 2f 9a 1b ec 12   ..).h<3.1L./....
    0090 - e4 fe 1b f7 c8 b5 fa 02-d7 a0 4f 22 09 95 ba 5a   ..........O"...Z

    Start Time: 1413441690
    Timeout   : 7200 (sec)
    Verify return code: 19 (self signed certificate in certificate chain)
---
closed
```

(3) 在 Nginx Server 上執行第一次 HTTPS 連線測試之後，馬上查驗 OCSP Server LOG 檔中，是否確實收到 OCSP 查詢之服務記錄，如下圖所示，原本 Access_log 中沒有任何資訊，於 14:41 確實有收到憑證狀態查詢之請求。

```
[root@RH5_GCAXCA logs]# date
? ? 10? ? 16 14:40:46 CST 2014
[root@RH5_GCAXCA logs]# cat GCA-access_log
[root@RH5_GCAXCA logs]# date
? ? 10? ? 16 14:41:10 CST 2014
[root@RH5_GCAXCA logs]# cat GCA-access_log
192.168.133.250 - - [16/Oct/2014:14:41:24 +0800] "GET /cgi-bin/OCSP2/ocsp_server.exe/MF
IwUDBOMEwwSjAJBgUrDgMCGgUABBSppW%2FTR5bLYRMmS8pM6twG1QAIHAQUeDzLPF7ya%2BkxtSlAS0SB%2BUG
2VZoCEQCwMOa7cDxBtYVTcNrbMp%2Bb HTTP/1.0" 200 1542
```

(4) 經過 4 分鐘左右由 Nginx Server 上執行”第二次”HTTPS 連線測試，因為第二次使用 HTTPS，就應該由 Nginx HTTP Server 本身的 OCSP Stapling 之功能，直接回復 OCSPResponse 回應訊息給予用戶端，如下圖所示，可以看到所 Cache 住的 OCSPResponse 內容，確實有出現 Next Update 資訊，且間格兩小時，也就是 Nginx 不需要再向 OCSP 服務詢問目前該 SSL 憑證狀態，直接拿取本身 Cache 住的 OCSPResponse 給予回應，如下所示：
時間點為 14:45 分左右，執行指令/usr/local/ssl/bin/openssl s_client -connect 192.168.133.250:443 -tls1 -tlsextdebug –status

```
[root@CentOS64 logs]# date
四 10月 16 14:44:51 CST 2014
[root@CentOS64 logs]# /usr/local/ssl/bin/openssl s_client -connect 192.168.133.250:443 -tls1 -tlsextdebug -status
CONNECTED(00000003)
TLS server extension "renegotiation info" (id=65281), len=1
0001 - <SPACES/NULS>
TLS server extension "session ticket" (id=35), len=0
TLS server extension "status request" (id=5), len=0
depth=2 C = TW, O = Test Root Certification Authority
verify error:num=19:self signed certificate in certificate chain
verify return:0
OCSP response:
==================================
OCSP Response Data:
    OCSP Response Status: successful (0x0)
    Response Type: Basic OCSP Response
    Version: 1 (0x0)
    Responder Id: C = TW, O = TL, OU = GCA, OU = OCSP Server, serialNumber = 0000000013018257
    Produced At: Oct 16 06:41:24 2014 GMT
    Responses:
    Certificate ID:
      Hash Algorithm: sha1
      Issuer Name Hash: A9A56FD34796CB6113264BCA4CEADC06D500081C
      Issuer Key Hash: 783CCB3C5EF26BE931B529404B4481F941B6559A
      Serial Number: B030E6BB703C41B5855370DADB329F9B
    Cert Status: good
    This Update: Oct 16 06:41:24 2014 GMT
    Next Update: Oct 16 08:41:24 2014 GMT
```

(5) 在 Nginx Server 上執行第二次 HTTPS 連線測試之後，馬上再查驗 OCSP Server LOG 檔中，是否確實收到 OCSP 查詢之服務記錄，如下圖所示，於 14:45 沒有收到任何的憑證狀態查詢之請求。

```
[root@RH5_GCAXCA logs]# date
? ? 10? ? 16 14:45:29 CST 2014
[root@RH5_GCAXCA logs]# cat GCA-access_log
192.168.133.250 - - [16/Oct/2014:14:41:24 +0800] "GET /cgi-bin/OCSP2/ocsp_server.exe/MF
IwUDBOMEwwSjAJBgUrDgMCGgUABBSppW%2FTR5bLYRMmS8pM6twG1QAIHAQUeDzLPF7ya%2BkxtSlAS0SB%2BUG
2VZoCEQCwMOa7cDxBtYVTcNrbMp%2Bb HTTP/1.0" 200 1542
```

(6) 經過約 8~9 分鐘左右由 Nginx Server 上執行"第三次"HTTPS 連線測試，因為第三次使用 HTTPS，一樣由 Nginx HTTP Server 本身的 OCSP Stapling 之功能，直接回復 OCSPResponse 回應訊息給予用戶端，如下圖所示，可以看到所 Cache 住的 OCSPResponse 內容，確實有出現 Next Update 資訊，且間格兩小時，也就是 Nginx 不需要再向 OCSP 服務詢問目前該 SSL 憑證狀態，直接拿取本身 Cache 住的 OCSPResponse 給予回應，如下所示：
時間點為 14:55 分左右，執行指令/usr/local/ssl/bin/openssl s_client -connect 192.168.133.250:443 -tls1 -tlsextdebug –status

```
[root@CentOS64 logs]# date
四 10月 16 14:54:51 CST 2014
[root@CentOS64 logs]# /usr/local/ssl/bin/openssl s_client -connect 192.168.133.250:443 -tls1 -tlsextdebug -status
CONNECTED(00000003)
TLS server extension "renegotiation info" (id=65281), len=1
0001 - <SPACES/NULS>
TLS server extension "session ticket" (id=35), len=0
TLS server extension "status request" (id=5), len=0
depth=2 C = TW, O = Test Root Certification Authority
verify error:num=19:self signed certificate in certificate chain
verify return:0
OCSP response:
==================================
OCSP Response Data:
    OCSP Response Status: successful (0x0)
    Response Type: Basic OCSP Response
    Version: 1 (0x0)
    Responder Id: C = TW, O = TL, OU = GCA, OU = OCSP Server, serialNumber = 0000000013018257
    Produced At: Oct 16 06:41:24 2014 GMT
    Responses:
    Certificate ID:
      Hash Algorithm: sha1
      Issuer Name Hash: A9A56FD34796CB6113264BCA4CEADC06D500081C
      Issuer Key Hash: 783CCB3C5EF26BE931B529404B4481F941B6559A
      Serial Number: B030E6BB703C41B5855370DADB329F9B
    Cert Status: good
    This Update: Oct 16 06:41:24 2014 GMT
    Next Update: Oct 16 08:41:24 2014 GMT
```

(7) 在 Nginx Server 上執行第三次 HTTPS 連線測試之後，馬上再查驗 OCSP Server LOG 檔中，是否確實收到 OCSP 查詢之服務記錄，如下圖所示，於 14:55 沒有收到任何的憑證狀態查詢之請求。

```
[root@RH5_GCAXCA logs]# date
? ? 10? ? 16 14:55:22 CST 2014
[root@RH5_GCAXCA logs]# cat GCA-access_log
192.168.133.250 - - [16/Oct/2014:14:41:24 +0800] "GET /cgi-bin/OCSP2/ocsp_server.exe/MF
IwUDBOMEwwSjAJBgUrDgMCGgUABBSppW%2FTR5bLYRMmS8pM6twG1QAIHAQUeDzLPF7ya%2BkxtSlAS0SB%2BUG
2VZoCEQCwMOa7cDxBtYVTcNrbMp%2Bb HTTP/1.0" 200 1542
```