1 Installation eines SSL Zertifikates unter Apache HTTP Server 1.3.x und 2.x (Dokumentenversion 1.4)

Table of Contents

1.1 Allgemeines

Dieses Dokument beschreibt den kompletten Ablauf zur Einbindung eines a.sign corporate SSL-Zertifikates in Ihren Apache 1.3.x und 2.x Webserver (http://www.apache.org). Sie haben dadurch die Möglichkeit, Ihre Homepage über eine sichere, verschlüselte Client-Server Verbindung aufzurufen und ein Abhören durch Dritte zu unterbinden.

1.2 Voraussetzungen

Um einen Apache 1.3.x oder 2.x Webserver SSL-fähig zu machen, benötigen Sie zusätzlich die zum Apache passende mod_ssl Version (http://www.modssl.org) zur Einbindung des SSL-Modules und openssl (http://www.openssl.org) zur Erzeugung eines Schlüsselpaares und des Certificate Requests. Auf Basis dieses Requests kann Ihnen a.trust ein a.sign corporate SSL Zertifikat ausstellen (Bestellung und Informationen auf http://www.a-trust.at/info.asp?node=243&lang=GE&ch=2).

1.3 Erstellen des Certificate Requests (PKCS#10) unter OpenSSL

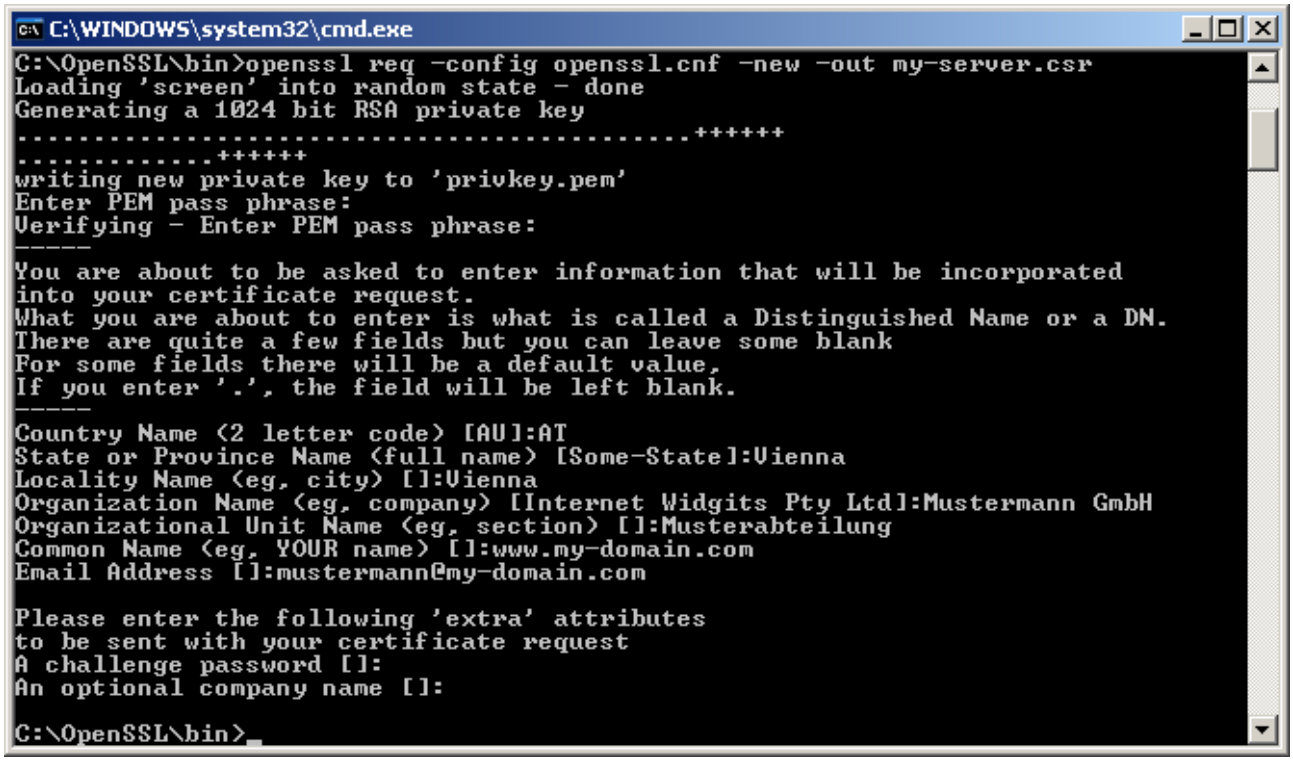
Bevor Sie mit der Erzeugung eines Schlüsselpaares und des Certificate Requests beginnen, kontrollieren Sie bitte, ob in Ihrer openssl-Konfigurationsdatei (openssl.cnf) die Schlüssellänge auf 1024 Bit gesetzt ist (eine Beispiel Konfigurationsdatei finden Sie auf unserer Homepage: http://www.a-trust.at/info.asp?node=169&lang=GE&ch=1&mch=2

Öffnen Sie nun die Eingabeaufforderung und wechseln Sie zu dem Pfad, in dem die Openssl Konfigurationsdatei abgelegt ist (z.B. C:\Openssl\bin). Geben Sie folgenden Befehl ein:

```
openssl req -config openssl.cnf -new -out my-server.csr
```

Mit diesem Befehl wird sowohl ein privater Schlüssel (privkey.pem – falls nicht anders in der openssl.cnf angegeben) als auch der öffentliche Request (my-server.csr) erzeugt. Sie müssen nun noch einige Angaben zum privaten Schlüssel tätigen, wie auf den folgenden

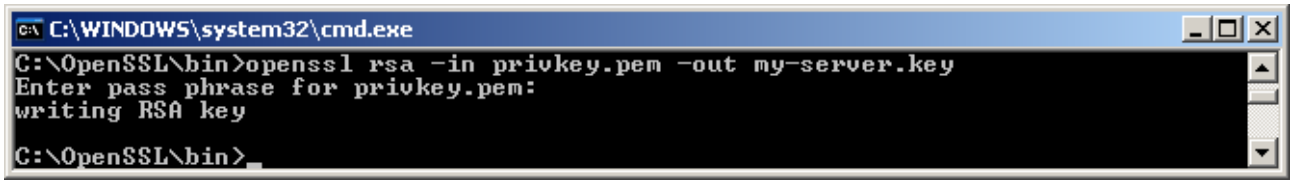
2 Masken sichtbar ist:



Die Angaben für Common Name (CN; muss Ihrem Domain Namen entsprechen, z.B. www.my-domain.com), Organisation (O) und Organisation Unit (OU) sind zwingend erforderlich, alle anderen Angaben sind optional.

Nun setzen Sie bitte noch folgenden Befehl ab:

```
openssl rsa -in privkey.pem -out my-server.key
```



Dieser Befehl entfernt das Passwort Ihres privaten Schlüssels und erzeugt ein key-File

(Sie werden dieses später bei der Implementierung des SSL-Zertifikates benötigen). Bitte bewahren Sei dieses File gut auf und geben Sie es niemandem weiter.

1.4 Senden des Requests an a.trust

Der zuvor erzeugte Certificate Request (my-server.csr) wird nun im Online-Bestell-formular eingefügt (einfach die Datei mit Notepad öffnen und den kompletten Inhalt kopieren).

Zur Kontrolle: Der Request sollte mit -----BEGIN CERTIFICATE REQUEST----- beginnen und mit -----END CERTIFICATE REQUEST----- enden.

1.5 Einbinden des SSL-Zertifikates unter der Version 1.3.x

Wichtiger Hinweis: Bevor Sie Änderungen an der Apache Konfiguration vornehmen, legen Sie bitte unbedingt eine Sicherheitskopie der httpd.conf-Datei an!

a.trust sendet Ihnen nach erfolgter Überprüfung die signierte Datei bin- und base64-encoded retour. Für den Apache Webserver benötigen Sie die base64-Datei (Dateiname lautet *_b64.crt, wobei * für Ihre Domain und der Signaturvertragsnummer steht). Erstellen Sie nun im Apache conf-Verzeichnis einen Ordner „ssl" und kopieren Sie die Dateien *_b64.crt und my-server.key in dieses Verzeichnis (legen Sie das key-File keinesfalls in das Verzeichnis htdocs!).

Nun muss noch die Apache Konfiguration erweitert werden, um eine SSL-Verbindung möglich zu machen. Öffnen Sie bitte die Konfigurationsdatei (httpd.conf) und fügen Sie noch folgende Zeilen hinzu:

```
LoadModule ssl_module modules/mod_ssl.so
Listen 80
Listen 443
SSLMutex sem
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
SSLSessionCache none
SSLLog logs/SSL.log
SSLLogLevel warn
<VirtualHost www.my-domain.com:443>
SSLEngine On
# Angabe und Ort des Server-Zertifikates
SSLCertificateFile conf/ssl/*_b64.crt
# Angabe und Ort des privaten Server-Schlüssels
SSLCertificateKeyFile conf/ssl/my-server.key
</VirtualHost>
```

Suchen Sie im httpd.conf File bitte noch nach einer Zeile namens Port 80 und ersetzen Sie sie durch # Port 80

Abschließend den Apache-Webserver neu starten und Sie sind am Ziel!

1.6 Einbinden des SSL-Zertifikates unter der Version 2.x

Die Einbindung des SSL-Zertifikates unter Apache 2.x verläuft ähnlich. Nur gibt es bei der Version 2.x eine eigene SSL-Konfigurationsdatei (ssl.conf).

Auch hier gilt: Bevor Sie Änderungen an der Apache Konfiguration vornehmen, legen Sie bitte unbedingt eine Sicherheitskopie der httpd.conf- und der ssl.conf-Datei an!

Öffnen Sie zuerst die httpd.conf Datei und überprüfen Sie ob das modssl Modul geladen ist. Suchen Sie nach der Zeile

```
LoadModule ssl_module modules/mod_ssl.so
```

Wenn ein # davor gesetzt ist, müssen Sie es entfernen, da sonst das mod_ssl Modul nicht geladen wird. Bei Apache 2.x sind die SSL-Einstellungen in einer eigenen Konfigurations-Datei ausgelagert – Sie müssen daher in der httpd.conf angeben, dass die SSL-Konfigurationsdatei ssl.conf mitgeladen wird - suchen Sie in der httpd.conf nach den Zeilen

```
<IfModule mod_ssl.c>
    Include conf/ssl.conf
</IfModule>
```

Sind diese beiden Einträge im httpd-conf File vorhanden, muss nun noch das ssl.conf File angepasst werden. Viele Befehle sind hier bereits vorgegeben und müssen nur noch aktiviert bzw. angepasst werden – die Befehle sind aber nahezu ident zu Apache 1.3.x. Suchen Sie sich folgende Zeilen heraus und passen Sie sie an Ihr SSL-Zertifikat an:

```
Listen 443
SSLMutex default
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
SSLPassPhraseDialog  builtin
SSLSessionCache dbm:logs/ssl_scache
SSLSessionCacheTimeout 300
<VirtualHost www.my-domain.com:443>
SSLEngine On
SSLCertificateFile conf/ssl/*_b64.crt
SSLCertificateKeyFile conf/ssl/my-server.key
</VirtualHost>
```

Abschließend den Apache2-Webserver neu starten und Sie sind am Ziel!

1.7 Informationen zu den SSL-Befehlen

Bei den unter Punkt 1.5 und 1.6 angegebenen SSL-Befehlen handelt es sich um Standardwerte, die errolgreich getestet wurden. Sie müssen nicht zwingend 1:1 in Ihre Apache Konfiguration übernommen werden.

Die genauen Definitionen der einzelnen Befehle können Sie auf http://www.modssl.org in der Online-Dokumentation nachschlagen.

1.8 SSL-Verbindung erzwingen

Optional haben Sie auch die Möglichkeit, für bestimmte Verzeichnisse eine SSL-Verbindung zu erzwingen (Aufruf der Seiten in diesem Verzeichnis ist dann nur über https möglich, nicht über http). In unserem Beispiel wurde der Apache Webserver im Verzeichnis C:/Programme installiert und es wurde ein Ordner „test" im htdocs-Startverzeichnis erstellt:

```
<Directory "C:/Programme/Apache/htdocs/test">
SSLRequireSSL
</Directory>
```

Soll der komplette Webinhalt ausschließlich über SSL aufgerufen werden, fügen Sie bitte noch folgende Zeilen ein:

```
<Directory />
SSLRequireSSL
</Directory>
```

1.9 Automatisch auf https umleiten

Sie haben aber auch die Möglichkeit, http-Seitenaufrufe automatisch auf https umzuleiten (wird z.B. http://www.my-domain.com angesurft, wird automatisch auf https://www.my-domain.com umgeleitet):

```
<VirtualHost www.my-domain.com:80>
Redirect permanent / https://www.my-domain.com/
</VirtualHost>
```

Oder aber Sie wollen nur ein bestimmtes Verzeichnis automatisch auf https umleiten (in unserem Beispiel wieder mit dem Verzeichnis „Test"):

```
<VirtualHost www.my-domain.com:80>
Redirect permanent /test https://www.my-domain.com/test
</VirtualHost>
```

Alle anderen Verzeichnisse können bei dieser Einstellung auch über eine gewöhnliche http-Verbindung aufgerufen werden.