1 Signierte Zertifikate

Zertifikate sind öffentliche Schlüssel, deren Echtheit von einer Zertifizierungsstelle durch eine *digitale Signatur* bestätigt wird.
Anstelle des umständlichen Ausdrucks *Zertifizierungsstelle*, wird im folgenden die Abkürzung des englischen Begriffs *Ceritfication Authority*, **CA**, verwendet.
Beim digitalen Signieren wird ein Prüfsummenwert (Hash) des öffentlichen Schlüssels mit dem privaten Schlüssel der CA verschlüsselt. Jeder kann nun mit dem öffentlichen Schlüssel der CA entschlüsseln und den Hashwert prüfen. Passt der Hashwert, ist auch die Echtheit des öffentlichen Schlüssels gewährleistet.
Der öffentliche Schlüssel der CA liegt bei kommerziellen CAs bereits auf dem Rechner bzw. im http-User-Agent (Browser, z.B. Firefox) vor.
Bei einer selbst erstellen CA muss deren öffentlicher Schlüssel in z.B. Firefox importiert werden.

2 Kurzanleitung zum Erstellen einer CA

2.1 openssl

Es wird vorausgesetzt, dass das Paket openssl installiert ist.
Ansonsten:

```
apt install openssl
```

2.2 PrivateKey der CA erstellen

aes256 verschluesselt den PrivateKey mit aes und passphrase (hier: ljwml!):

```
openssl genrsa -aes256 -out ca-key.pem 2048
```

2.3 Öffentichen Schlüssel der CA erstellen

Der öffentliche Schlüssel der CA wird root-certificate genannt.

```
openssl req -x509 -new -nodes -extensions v3_ca -key ca-key.pem \
-days 3650 -out ca-root.pem -sha512
```

Das Ergebnis anschauen:

```
openssl x509 -text -noout -in ca-root.pem
```

2.4 Importieren des root-certificates ins System

Das root-certificate kann unter Ubuntu/Debian wie folgt importiert werden. Dazu sind root-rechen erforderlich. Achtung, die Dateiendung muss **.crt** sein:

```
cp ca-root.pem /usr/share/ca-certificates/fti2tCA.crt
dpkg-reconfigure ca-certificates
```

Fertig ist die eigene CA (certification authority).

2.5 Installation ins Apache-Konfig-Verzeichnis

1. Verzeicnis `/etc/apache2/ssl.crt` erstellen
2. ca-root.pem nach `/etc/apache2/ssl.crt/` kopieren

3 Eigene Zertifikate ausstellen

Im Folgenden wird vorausgesetzt, dass die bereits erstellten Schlüssel der CA wie in folgender Verzeichnisstruktur abgelegt sind. Das Verzeichnis apache2 ist zunächst leer und ist das Arbeitsverzeichnis. Es wird nach und nach mit den schon hier angebenen Dateien gefüllt:

```
.
|--- apache2
|     |-- schuleMachtSpass.csr
|     |-- schuleMachtSpass-key.pem
|     +-- schuleMachtSpass-ssl-cert.pem
|
+--- CA
      |-- ca-key.pem
      |-- ca-root.pem
      +-- passphrase.txt
```

3.1 Privater Schluessel

```
openssl genrsa -out schuleMachtSpass-key.pem 2048
```

3.2 Certifikate Signing Request erstellen

```
openssl req -new -key schuleMachtSpass-key.pem \
-out schuleMachtSpass.csr -sha512
```

3.3 Das eigentiche Zertifikat signieren

ACHTUNG, ACHTUNG: der Common Name (CN) MUSS dem Domain-Name des virtual Hosts entsprechen.

Hier also: **schule.macht.spass**

```
openssl x509 -req -in schuleMachtSpass.csr -CA ../CA/ca-root.pem \
-CAkey ../CA/ca-key.pem -CAcreateserial \
-out schuleMachtSpass-ssl-cert.pem -days 365 -sha512
```

3.4 Privater Schlüssel und Zertifikat kopieren

Der private Schlüssel und das Zertifikat können nun in die entsprechenden Verzeichnisse kopiert werden:

```
/etc/ssl/certs
/etc/ssl/private
```

4 Apache2 Beispielkonfiguration

Damit SSL funktioniert, muss das Modul ssl für Apache2 aktiviert werden:

```
a2enmod ssl
```

Beispielkonfig:

```
<IfModule mod_ssl.c>
      <VirtualHost *:443>
              ServerAdmin webmaster@localhost

              DocumentRoot /var/www/schule
              ServerName   schule.macht.spass

              ErrorLog ${APACHE_LOG_DIR}/error.log
              CustomLog ${APACHE_LOG_DIR}/access.log combined

              SSLEngine on

              # pfadangaben zu den selbsterstellen zertifikaten
              SSLCertificateFile /etc/ssl/certs/schuleMachtSpass-ssl-cert.pem
              SSLCertificateKeyFile /etc/ssl/private/schuleMachtSpass-key.pem

              #   Certificate Authority (CA):
              #   Set the CA certificate verification path where to find CA
              #   certificates for client authentication or alternatively one
              #   huge file containing all of them (file must be PEM encoded)
              #   Note: Inside SSLCACertificatePath you need hash symlinks
              #                 to point to the certificate files. Use the provided
              #                 Makefile to update the hash symlinks after changes.
              #SSLCACertificatePath /etc/ssl/certs/
              SSLCACertificateFile /etc/apache2/ssl.crt/ca-root.pem

              <FilesMatch "\.(cgi|shtml|phtml|php)$">
                              SSLOptions +StdEnvVars
              </FilesMatch>
              <Directory /usr/lib/cgi-bin>
                              SSLOptions +StdEnvVars
              </Directory>


      </VirtualHost>
</IfModule>
```

5 Weiterleitung Port 80 auf Port 443

Damit Benutzer nur über ssl auf den Server zugreifen können, kann man in der Datei `000-default.conf` die Zeile

```
Redirect / https://schule.macht.spass
```

eintragen. Dann werden Zugriffe auf Port 80 auf Port 443 umgeleitet.