Apache2: SSL-verschlüsselte Verbindungen ermöglichen (Ubuntu 14.04)

Seit dem Bekanntwerden der flächendeckenden, verdachtslosen Überwachung durch die NSA und andere Geheimdienste gewinnt Verschlüsselung im Internet an Bedeutung. Vor allem die Kommunikation zwischen Browser und Webserver wird verstärkt gesichert, um privates privat zu halten. Eine solche SSL-Verschlüsselung kann auch für den eigenen Apache Server eingerichtet werden. Benötigt wird dafür das Apache Modul „ssl“:

```
a2enmod ssl
service apache2 restart
```

Da die verschlüsselte Kommunikation zwischen Browser und Webserver nicht über Port 80, sondern über Port 443 abläuft, muss für jeden Port-80 -VirtualHost eine Entsprechung mit Port 443 angelegt werden, wenn SSL für eine (Sub-/)Domain verwendet werden soll:

```
<VirtualHost *:443>
    ServerName beispiel.de
    DocumentRoot /var/www/beispiel.de/

    SSLEngine on
    SSLCertificateFile      /etc/ssl/certs/ssl-cert-snakeoil.pem
    SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key
#   SSLCertificateChainFile /etc/apache2/ssl.crt/server-ca.crt
</VirtualHost>
```

Der untere Teil der Konfiguration legt die SSL-Einstellungen fest. Die SSL-Engine wird aktiviert und die Pfade zu den Zertifikaten festgelegt. Benötigt werden mindestens ein Public Key (CertificateFile) und ein Private Key (CertificateKeyFile). Die Zeile der dritten Pfadangabe ist auskommentiert. Sie wird häufig benötigt, wenn Zertifikate von anerkannten Zertifikatsanbietern wie z.B. StartSSL verwendet werden sollen. Aktuell werden aber keine offiziell anerkannten Zertifikate genutzt, sondern Testzertifikate, die vom Server selbst ausgestellt wurden. Die letzte Zeile wird daher nicht benötigt. Diese sind übrigens nicht vertrauenswürdig, weshalb die Browser später warnen werden, wenn eine verschlüsselte Verbindung hergestellt wird. Für Testzwecke reicht ein selbst erzeugtes Zertifikat aber zunächst aus.

SSL Sicherheit verstärken

Um größtmögliche Sicherheit zu erreichen, soll der Browser der Nutzer angewiesen werden, die stärksten verfügbaren Verschlüsselungsalgorithmen zu nutzen und erst auf schwächere auszuweichen, wenn keine stärkeren verfügbar oder möglich sind. Dazu wird der VirtualHost um folgende Zeilen erweitert:

```
SSLProtocol All -SSLv2 -SSLv3
SSLCompression off
SSLHonorCipherOrder On
```

```
SSLCipherSuite
EDH+CAMELLIA:EDH+aRSA:EECDH+aRSA+AESGCM:EECDH+aRSA+SHA384:EECDH+aRSA+SHA256:
EECDH:+CAMELLIA256:+AES256:+CAMELLIA128:+AES128:+SSLv3:!aNULL:!eNULL:!LOW:!3
DES:!MD5:!EXP:!PSK:!DSS:!RC4:!SEED:!ECDSA:CAMELLIA256-SHA:AES256-
SHA:CAMELLIA128-SHA:AES128-SHA
Header add Strict-Transport-Security "max-age=15768000"
```

(Siehe auch: „Appied crypto hardening - bettercrypto.org„)

Nach den Änderungen an der Konfiguration muss Apache neu geladen werden: <code bash> service apache2 reload </bash> Der VirtualHost sollte nun auch über https erreichbar sein. Euch wird im Browser beim ersten Aufruf eine Warnung angezeigt, die ihr aber ignorieren könnt. Für den Einsatz von SSL in der Öffentlichkeit sollten Zertifikate von anerkannten SSL-Certificate Authorities verwendet werden, damit Besucher eurer Seite nicht durch Warnungen abgeschreckt werden. Diese sind normalerweise nicht besonders günstig. Eine Ausnahme ist hier startssl.com – die verteilen einfache Zertifikate auch kostenlos.