Wolfgang Kirk

Good Practice - 34: Server mit Transport Layer Security sichern (OpenSSL)

1 Vorbemerkung

Wer einen Server für seinen Dateitransfer (*File Transfer Protocol*, FTP) und / oder einen Webserver (Apache, HTTP) installiert hat, weiß auch, dass damit nicht automatisch alle Fragen der IT-Sicherheit beantwortet sind. Im Regelfall ist keine sichere Verbindung eingerichtet. So auch bei den Grundinstallationen von Vsftpd-Server und Ubuntu-Server. Das heißt, dass der interne und externe Datenaustausch über die Leitungen unverschlüsselt erfolgt.

Abhilfe schafft hier die Einrichtung von sicheren Verbindungen mittels *Transport Layer Security* (TLS). Das veralterte Protokoll SSL wird heute nicht mehr unterstützt, da es unsicher ist. Die Bezeichnung SSL ist heute ein Synonym für TSL. In vielen Veröffentlichungen wird deshalb auch die Bezeichnung SSL/TSL benutzt. Einen Überblick über die Versionsgeschichte bietet Wikipedia:
URL: **https://de.wikipedia.org/wiki/Transport_Layer_Security** [2021-05-03].

2 Voraussetzungen

Vorausgesetzt werden:

- installierter Vsftpd-Server (vgl. Kirk, Wolfgang: **Tipps: Vsftpd einrichten unter Ubuntu**)
- der Dateiaustausch mittels FTP erfolgt nur im Heimnetz (intern),
- installierter Webserver (vgl. Kirk, Wolfgang: **Good Practice - 21: Eigenen Webserver einrichten (Linux)**)
- Router: Port-Weiterleitung zum FTP-Server für Port 20 sowie Webserver für Port 80 und 443 (*Port-Forwarding*) ist eingerichtet
- Firewall: Die Ports 80 und 443 sind für Apache freigegeben.

Die Installationsbeschreibung betrifft Debian-basierte Linux-Systeme, vornehmlich die Distribution Ubuntu und deren Ableger. In anderen Linux-Distributionen ist die Installation jedoch grundsätzlich ähnlich.

Standardmäßig ist das kostenfreie OpenSSL bereits in Ubuntu enthalten. Wenn dies nicht der Fall ist, führen Sie die folgende Befehlszeile in einem Terminal aus:

```
sudo apt-get install openssl
```

Weiterführende Informationen zu OpenSSL bietet die Seite von Let's Encrypt: URL: **https://letsencrypt.org/de/** [2021-05-03].

3 Sicherer Dateiaustausch (ftps)

Die Absicherung mit TLS erfolgt in folgenden Schritten:

Schritt 1: Konfiguration aktualisieren

Die Konfigurationsdatei wird geöffnet:

```
sudo nano /etc/vsftpd.conf
```

Die Einträge **rsa_cert_file** und **rsa_private_key_file** sowie **ssl_enable** sind wie folgt anzupassen

```
rsa_cert_file=/etc/ssl/certs/vsftpd.pem
rsa_private_key_file=/etc/ssl/private/vsftpd.key
ssl_enable=YES
```

Die Änderungen speichern und den Editor verlassen.

Schritt 2: Schlüssel und Zertifkat generieren

Der *private Schlüssel* wird erstellt mit:

```
sudo openssl genrsa -out /etc/ssl/private/vsftpd.key
```

Sie werden aufgefordert, einige Informationen wie Ihr Land, Ihre Stadt, Ihre E-Mail-Adresse usw. anzugeben. Bitte lesen Sie die Anweisungen sorgfältig durch.

Die *Zertifizierungsanforderung* für den vorstehend erstellten Schlüssel wird mit folgendem Befehl generiert:

```
sudo openssl req -new -key /etc/ssl/private/vsftpd.key -out /etc/ssl/certs/vsftpd.csr
```

Jetzt kann das für ein Jahr gültige *Zertifkat* erstellt werden:

```
sudo openssl x509 -req -days 365 -in /etc/ssl/certs/vsftpd.csr -signkey /etc/ssl/private/vsftpd.key -out /etc/ssl/certs/vsftpd.pem
```

Schritt 3: Vsftpd neu starten

```
sudo systemctl restart vsftpd
```

Schritt 4: Verbindung testen

Wenn mit FileZilla eine Verbindung aufgebaut wird, erscheint ein Bildschirm, mit dem Hinweis, dass eine TLS-Verbindung hergestellt werden konnte.

Schritt 5: Zertfikat dauerhaft vertrauen

Damit nicht bei jedem Aufruf der Hinweisbildschirm erscheint, wählen Sie die Option, diesem Zertifikat zukünftig immer zu vertrauen. Klicken Sie dann auf OK, um mit der Verbindung fortzufahren.

Hinweis

> Für den verschlüsselten internen Dateiaustausch reicht ein selbst erstelltes Zertifikat.

Nutzung des Dateimanagers

Man kann auch mit dem Dateimanager, z.B. Thunar, eine ftps-Verbindung herstellen:

Mit *STRG-L* eine Befehlszeile öffnen und eingeben:

```
ftps://XXX.XXX.X.XXX:20/tld
```

Die IP-Adresse ist den eigenen Verhältnissen anzupassen. Im Regelfall ist der Port 20 für die ftp-Verbindung freigegeben.

Nach dem Aufruf erscheint ein Hinweis *Überprüfung der Identität fehlgeschlagen*. Klicken Sie *OK* um fortzufahren. Danach folgt der Hinweis zur Eingabe der Zugangsdaten.

Weitere Informationen siehe knetfeder.de: Beitrag FTPS-Zugriffe mit dem Dateimanager unter Linux [Stand: 2018-09-17], URL: **https://www.knetfeder.de/linux/index.php?id=221**.

4 Sicherer Webserver (https)

Hier wird die Zertifizierung in der Umgebung mit Apache-Server beschrieben. Vor der Einrichtung von TLS sollte der Status ermittelt werden:

```
systemctl status apache2.service
```

4.1 Konfigurationsdatei bearbeiten

Die Datei mit einem Editor öffnen:

```
sudo nano /etc/apache2/sites-avialabe/000-default.conf
```

Folgende Eintragungen vornehmen:

Im Abschnitt <VirtualHost *:80>

ServerName wunschname.de
ServerAlias **www.wunschname.de**

Zusätzlich ist der VirtualHost 443 einzutragen:

<VirtualHost *:443>
ServerName wunschname.de
ServerAlias **www.wunschname.de**
`</VirtualHost>`

Die Angabe *wunschname.de* ist den eigenen Verhältnissen entsprechend anzupassen.

Damit die Änderungen wirksam werden, den Server neu starten:

```
systemctl restart apache2
```

4.2 SSl-Modul starten

Im Terminal ist folgendes auszuführen:

```
sudo a2enmod ssl
```

```
sudo a2ensite default-ssl
```

```
sudo a2enmod rewrite
```

```
systemctl restart apache2
```

4.2 Einrichtung

Schritt 1: ServerName global festlegen

Zur globalen Vergabe des ServerName wird die Konfigurationsdatei von Apache geöffnet:

```
sudo nano /etc/apache2/apache2.conf
```

In die erste Zeile wird eingetragen bzw. aktualisiert:

ServerName wunschname.de

Die Angabe *wunschname.de* ist den eigenen Verhältnissen entsprechend anzupassen.

Schritt 2: certbot installieren

Das Repository in der Paketverwaltung eintragen:

```
sudo add-apt-repository ppa:certbot/certbot
```

```
sudo apt-get update
```

Das Programm wird installiert mit folgendem Befehl:

```
sudo apt-get install python3-certbot-apache
```

Dies gilt für die Ubuntu Version 21.04.

Für ältere Versionen kann auch der folgende Befehl ausgeführt werden:

```
sudo apt-get install python-certbot-apache
```

Schritt 3: Öffentliches Zertifikat erstellen

Das Zertifikat wird für die global festgelegte *wunschname.de*-Domain erstellt:

```
sudo certbot --apache -d wunschname.de
```

Mit Angabe von www:

```
sudo certbot --apache -d www.wunschname.de
```

Die Angabe *wunschname.de* ist den eigenen Verhältnissen entsprechend anzupassen.

Danach werden folgende Schritte ausgeführt:

- Geben Sie eine E-Mail-Adresse für Erneuerungs- und Sicherheitshinweise ein
- Stimmen Sie den Nutzungsbedingungen zu
- Geben Sie an, ob E-Mails von EFF empfangen werden sollen
- Wählen Sie, ob der HTTP-Verkehr an HTTPS umgeleitet werden soll -
  - 1 (keine Umleitung, keine weiteren Änderungen am Server) oder
  - 2 (alle Anforderungen an HTTPS umleiten)

Gewollt ist eine https-Verbindung: Also 2 auswählen und bestätigen.

Wenn die Nachricht „Congratulations! You have successfully [...]“ erscheint, ist das Zertifikat erfolgreich installiert.

Schritt 4: Automatische Erneuerung

Die so erstellten Zertifikate werden regelmäßig nach 90 Tagen ungültig. Viele Distributionen haben standardmäßig automatische Verlängerungen aktiviert, entweder über Systemzeitgeber oder Cron-Jobs.

- Einstellungen anzeigen
  - unter systemd:

    `systemctl show certbot.timer`

  - Einstellungen anzeigen unter anderen Systemen

    `cat /etc/cron.d/certbot`

  - Test von *certbot renew*

    `sudo certbot renew --dry-run`

- Ausführung von *certbot renew* prüfen

  `systemctl list-timers`

  `ls /etc/cron*`

Sollte *certbot renew* noch nicht ausgeführt werden, kann ein selbststartendes *cert-renew-script* installiert werden (optional) :

```
crontab -e
```

Beim erstmaligen Aufruf 1 auswählen, damit der Nano-Editor gestartet wird.

Am Ende von crontab folgendes einfügen:

```
2 0 * * * certbot renew --post-hook "systemctl restart apache2"
```

Damit wird jede Nacht um 2 Uhr der Befehl „certbot renew“ ausgeführt und Apache neu gestartet.

Schritt 5: Sperren / Widerruf des Zertifikates

```
certbot revoke --cert-name cert_name
```

oder

```
certbot revoke --cert-path /path/to/cert.pem
```

Die Angaben sind mit den gültigen Systemdaten zu ersetzen.

5 Apache-Konfiguration

Vor einem Neustart sollte man unbedingt die Konfigurations-Datei auf Fehler überprüfen:

```
sudo apachectl configtest
```

Wenn der Test mit *OK* angezeigt wird, kann der Webserver neu gestartet werden:

```
systemctl restart apache2
```

Jetzt kann die eigene Domain mit **https://wunschname.de** aufgerufen werden und die Startseite sollte erscheinen. Die Angabe *wunschname.de* ist den eigenen Verhältnissen entsprechend anzupassen.

Impressum

ISBN 978-3-96619-152-4 (PDF)
ISSN 2627-8758
GUID 10ffc0b8-a30e-4578-aa3d-52fccb40cfc2

ISNI 0000000459074303
ORCID ID https://orcid.org/0000-0002-2359-6164
Blog https://wolfgangkirk.de

Der Text ist als Band 81 Teil von Veröffentlichungen in der Reihe *Digitale Gesellschaft in Deutschland* (ISSN 2627-8758 elektronische Publikationen).

Textsatz mit Typora in Markdown und mit Pandoc in das Zielformat konvertiert.

Stand: 2021-05-03

Haftungsausschluss

Der Autor haftet insbesondere nicht für den Inhalt der vorgestellten Internet-Seiten. Die Verantwortung für Inhalt und Funktion der Links liegt bei den jeweiligen Betreibern. Rechtswidrige Inhalte waren zum Zeitpunkt der Verlinkung nicht erkennbar.

Lizenz

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie. Detaillierte bibliografische Daten sind im Internet über https://dnb.de abrufbar.

BibTeX