

# Bereitstellung von Stirling-PDF

Dieser Abschnitt beschreibt, wie Stirling-PDF normalerweise aufgerufen wird und wie sich dies optimieren lässt.

- [Zugang zur Anwendung](#)
- [Ausgangslage und Zielsetzung](#)
- [Zugang per ReverseProxy](#)

# Zugang zur Anwendung

Out-of-the-Box ist Stirling-PDF nach dem Schema *HTTP://IP-ADRESSE:PortNummer* erreichbar,  
konkret bedeutet dies bspw.:  
bei IPv4 `http://1.2.3.4:8080`  
bei IPv6 `http://[2:3:4:5:a:b:c:d]:8080`

Die Bereitstellung von Diensten per unverschlüsseltem HTTP ist unzeitgemäß und ein Sicherheitsrisiko.

Außerdem ist der Port 80 (Standard-Port für HTTP) häufig bereits durch einen (anderen) Webserver belegt und alternative Ports (hier 8080) können häufig aufgrund restriktiver Firewall-Regeln bspw. in Unternehmensnetzwerken nicht erreicht werden.

# Ausgangslage und Zielsetzung

Das hier behandelte Szenario basiert auf folgender Infrastruktur:

- Zur Verfügung stehen eine [IPv4](#)-Adresse und ein [IPv6](#)-Subnet (Standard 64-Bit).  
Die [IPv4](#)-Adresse ist einem [Proxmox](#)-Server zugewiesen, der diverse virtuelle Maschinen beherbergt, die sich in einem separaten Netzwerk ([Brücke mit NAT](#)) befinden.  
Das [IPv6](#)-Subnet wird ebenfalls vom [Proxmox](#)-Server verwaltet und über [geroutete Brücken](#) an die virtuellen Maschinen verteilt.
- Eine dieser virtuellen Maschinen stellt einen Webserver [Apache](#) 2.4 als [ReverseProxy](#) zur Verfügung.  
Der Traffic, der beim Proxmox-Host auf den Ports 80 (HTTP) und 443 (HTTPS) ankommt, wird direkt zu diesem [ReverseProxy](#) durchgeleitet.
- [Stirling-PDF](#) läuft auf einer weiteren virtuellen Maschinen.
- Die VM mit [Stirling-PDF](#) ist nicht direkt von außen zu erreichen:  
Zwar könnte sie über eine [IPv6](#)-Adresse aus dem bereit gestellten Subnetz angesprochen werden -  
aber damit wären all die Nutzer ausgeschlossen, die nur über eine [IPv4](#)-Anbindung verfügen.  
Eine weitere (individuelle) [IPv4](#)-Adresse (nur für [Stirling-PDF](#)) scheidet aus Ressourcengründen (Kosten und Verfügbarkeit) aus.  
[Stirling-PDF](#) wird über Port 8080 bereit gestellt -  
dieser ist jedoch aus manchen Netzwerken nicht zu erreichen, weil insbesondere Unternehmensnetzwerke häufig sehr restriktive Firewall-Regeln haben und häufig nur HTTPS via Port 443 erlauben.  
Eine Port-Weiterleitung von Port 80 am Host auf Port 8080 zur VM mit [Stirling-PDF](#) scheidet aus, weil Port 80 am Host bereits anderweitig belegt ist.

Ziel ist es, die Anwendung [Stirling-PDF](#) per sicherem HTTPS über den Standard-Port (443) zur Verfügung zu stellen.

# Zugang per ReverseProxy

Die Konfiguration eines [ReverseProxy](#) und der Einsatz von [Let's Encrypt](#) werden in einem separaten Buch aufgegriffen.

Hier an dieser Stelle daher nur eine kurze Zusammenfassung.

Auf dem [ReverseProxy](#) (hier ein [Apache 2.4](#) Web-Server) wird für dieses Beispiel folgende Konfiguration (bspw. als `/etc/apache2/sites-enabled/stirling.conf`) angelegt:

```
<VirtualHost subdomain.schubert.si:80>
    ServerAdmin webmaster@schubert.si
    ServerName subdomain.schubert.si
    ServerSignature Off
    DocumentRoot /var/www/html/stirling
    RewriteEngine on
    RewriteCond %{SERVER_NAME} =subdomain.schubert.si
    RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R=permanent]
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

Das unter *DocumentRoot* angegebene Verzeichnis ist nur ein Dummy mit einer leeren *index.html*-Datei.

Gebraucht wird das leere Verzeichnis nur für den Einsatz des Certbot von [Let's Encrypt](#).

Auch wenn es zunächst so aussehen mag:

Es wird kein Dienst unter Port 80 (unverschlüsseltes HTTP) angeboten, sondern per Rewrite wird jeglicher Aufruf von HTTP auf HTTPS umgeschrieben.

Analog zur Konfiguration für Anfragen an Port 80 gibt es eine weitere Konfiguration (bspw. als `/etc/apache2/sites-enabled/stirling-le-ssl.conf`) oder als weiteren Abschnitt in der ersten Konfigurationsdatei, um eben die Anfragen an Port 443 (HTTPS) zu beantworten:

```
<IfModule mod_ssl.c>
<VirtualHost subdomain.schubert.si:443>
    ServerAdmin webmaster@schubert.si
    ServerName subdomain.schubert.si
    ServerSignature Off
    ProxyPreserveHost On
```

```
ProxyPass / http://192.168.1.11:8080/
ProxyPassReverse / http://192.168.1.11:8080/
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
SSLCertificateFile /etc/letsencrypt/live/subdomain.schubert.si/fullchain.pem
SSLCertificateKeyFile /etc/letsencrypt/live/subdomain.schubert.si/privkey.pem
Include /etc/letsencrypt/options-ssl-apache.conf
</VirtualHost>
</IfModule>
```

Hinter *ProxyPass...* wird die (interne) IPv4-Adresse der virtuellen Maschine angegeben, die Stirling-PDF bereitstellt.

Dafür müssen sich natürlich sowohl die VM mit dem [ReverseProxy](#) als auch die VM mit Stirling-PDF im gleichen Subnetz befinden.

Sollte der [ReverseProxy](#) auf der gleichen Maschine laufen wie Stirling-PDF selbst, wird stattdessen localhost (<http://localhost:8080> oder <http://127.0.0.1:8080>) verwendet

Nach Anlage der Konfiguration muss diese Apache noch mittels `systemctl reload apache2` bekannt gemacht werden.

Somit kann Stirling-PDF per HTTPS unter dem Standard-Port 443 anstatt über eine unverschlüsselte HTTP-Verbindung genutzt werden.