Konfiguration des PicApport Servers für SSL mit Letsencrypt

Hintergrund Information

Der Offline Modus von PicApport wird von modernen Browsern nur noch für SSL Verbindungen unterstützt. Stichwort hierzu ist "*progressive Web App*" (PWA).

PicApport nutzt hierfür je nach Konfiguration entweder:

- Application Cache (AppCache)
- oder die moderneren "Service Worker"

Der Aufwand und die Kosten für Privatanwender und kleine Firmen Server über DynDns mit einem gültigen Zertifikat auszustatten ist relativ hoch.

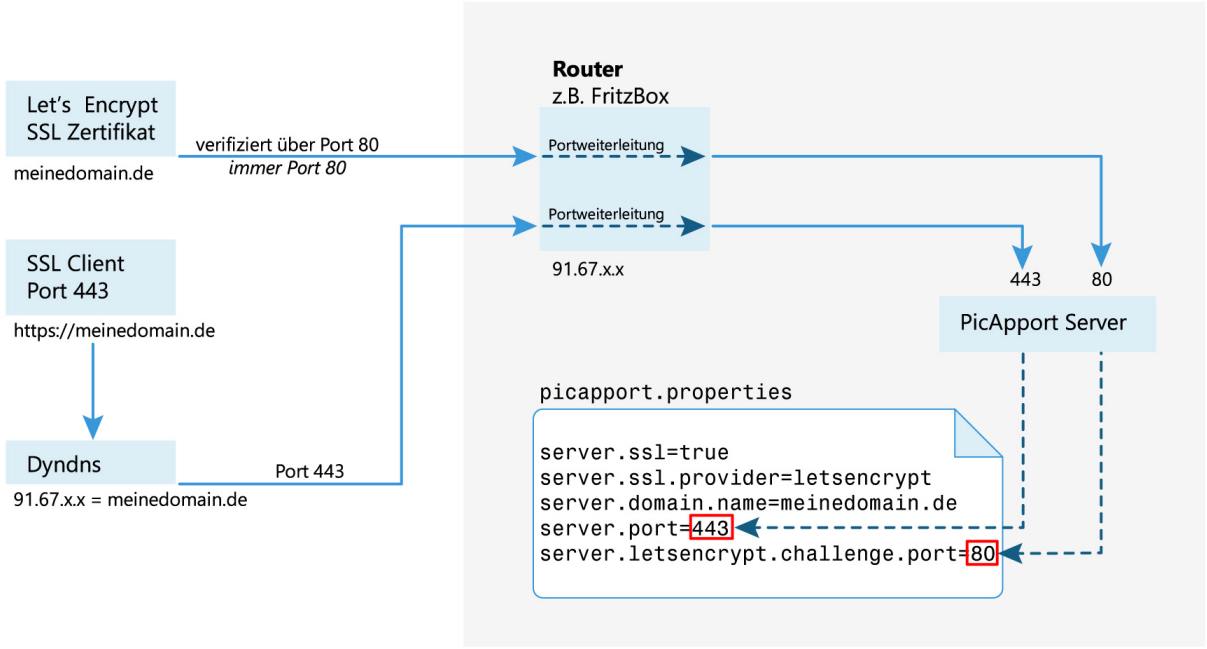
https://letsencrypt.org/ bietet hier eine Lösung über das standardisierte ACME Protokoll welches wir ab Version 7.6.x in PicApport implementiert haben.

Ziel hierbei ist:

- SSL einmalig in PicApport für Letsencrypt zu konfigurieren
- Einmal konfiguriert sorgt PicApport automatisch für die Aktualisierung der Zertifikate
  Alles funktioniert voll automatisch.

Konfiguration

Das folgende Schaublid zeigt wie SSL für Letsencrypt konfiguriert wird:



Wie im obigen Bild ersichtlich müssen 5 Parameter in die ***picapport.properties*** eingetragen werden. Nach einem Neustart des Servers sollte alles automatisch eingerichtet werden.

Die Letsencrypt-Aktivitäten werden in den Logdateien unter *de.contecon.picapport.security.utils.LetsEncryptService::* dokumentiert (ab Version 7.6 auch über die Weboberfläche abfragbar wenn man berechtigt ist)

```
MSG  @ 02:53:53.040 de.contecon.picapport.security.utils.LetsEncryptService:: OK: valid certificate found. No
renew necessary.
MSG  @ 02:53:55.008 de.contecon.picapport.security.utils.LetsEncryptService:: UPDATE: certificate expired.Tue
Jul 09 02:53:55 CEST 2019-Mon Oct 07 02:53:55 CEST 2019
MSG  @ 02:53:55.008 de.contecon.picapport.security.utils.LetsEncryptService:: UPDATE: no matched entries in
keystore found
MSG  @ 02:53:55.008 de.contecon.picapport.security.utils.LetsEncryptService:: UPDATE: starting renew
MSG  @ 02:53:56.571 de.contecon.picapport.security.utils.LetsEncryptService:: UPDATE: challenge accepted
MSG  @ 02:53:59.008 de.contecon.picapport.security.utils.LetsEncryptService:: OK: challenge has been completed
```

Wichtiger Hinweis

server.port sowie ***server.letsencrypt.challenge.port*** können frei gewählt werden.
Allerdings muss sichergestellt sein, das der Challenge-Port von "außen" immer über Port 80 erreicht werden kann.

Das ist eine Vorgabe von Letsencrypt und kann nicht verändert werden.

Bitte auch beachten das Portnummern < 1024 unter Linux (incl. Apple) sogenannte ***Priviliged ports*** sind und entsprechend behandelt werden müssen.
siehe hierzu auch: https://stackoverflow.com/questions/413807/is-there-a-way-for-non-root-processes-to-bind-to-privileged-ports-on-linux