Zertifikate erstellen

> Alle nachfolgenden, erforderlichen Eingaben, Kommandos, etc. sind im Terminal des PI OS erforderlich!
>
> Je nach dem welche Anleitung ihr im Internet schon gelesen habt, werden immer verschiedene Verzeichnisse zur Ablage der Zertifikate vorgeschlagen oder verwendet. Ich würde euch vorschlagen ein Verzeichnis zu erstellen, das ihr immer verwendet um eure eigenen Zertifikate zu erstellen, zu bearbeiten und als Speicherort in den Browser 'VirtualHost' Blöcken anzugeben.
>
> Für unser Beispiel verwenden wir:
>
> - TLD = lan
> - SLD = zuhause
> - subdomain = nas

1. erstellen einer vertrauenswürdigen Stammzertifizierungsstelle

1.1 erstellen eines Privaten Schlüssels für die Stammzertifizierungsstelle

Kommando: "openssl genrsa -des3 -out <SLD, ohne TLD>.key 2048"

- z.B. euer Domänenname = zuhause.lan, dann nur "zuhause"!
- ihr werdet dann nach einer "Pass Phrase" gefragt **8-tung!** vergebt bitte eine Passwort, dass ihr **nicht vergesst**!, da ansonsten später massive Probleme auftauchen können, da ihr dieses Passwort immer mal wieder braucht, je nachdem wo und wie ihr euer Zertifikat gebrauchen wollt!
- > *noch ein Tipp*:
  > - je länger das Passwort, desto besser die Verschlüsselung. Obwohl ich auch nicht übertreiben würde, da ja sich alles nur in eurem Heimnetzwerk abspielt.
  > - übrigens: falls ihr im Laufe der Arbeiten feststellt, das ihr einen anderen TLD oder SLD benutzen wollt, müsst ihr immer wieder von vorne anfangen. Zertifikate kann man nicht ändern!

1.2 erstellen des Stammzertifikats

- Kommando: "openssl req -x509 -new -nodes -key <SLD>.key -sha256 -days 3650 -out >SLD>.pem"
- es werden jetzt noch folgenden zusätzliche Informationen abgefragt:
-

> “
> Country Name (2 letter code): <>
> State or Province Name (full name): <>
> Locality Name (eg, city): <>
> Organization Name (eg, company): <>
> Organizational Unit Name (eg, section): <>
> Common Name (e.g. server FQDN or YOUR name): <SLD.TLD>
> Email Address: <>

<> = könnt ihr ausfüllen, muss man aber nicht!

2. erstellen von Zertifikaten basierend auf dem Stammzertifikat

> “ Diesen Ablauf müsst ihr für jede subdomain (meistens Geräte, z.b. NAS) wiederholen!

2.1 erstellen eines Privaten Schlüssels für das Zertifikat

Kommando: "openssl genrsa -out <subdomain.SLD.TLD>.key 2048"

- unser Beispiel = nas.zuhause.lan

2.2 erstellen eines CSR (**C**ode **S**igning **R**equest)

Kommando: "openssl req -new -key <subdomain.SLD.TLD>.key -out <subdomain.SLD.TLD>.csr"

- unser Beispiel = nas.zuhause.lan

2.3 erstellen einer X509 V3 Zertifikatserweiterungs-Konfigurationsdatei

Kommando: "nano <subdomain.SLD.TLD>.ext"

```
authorityKeyIdentifier=keyid,issuer
basicConstraints=CA:FALSE
keyUsage = digitalSignature, nonRepudiation, keyEncipherment, dataEncipherment
subjectAltName = @alt_names

[alt_names]
```

```
DNS.1 = <subdomain.SLD.TLD>
```

- unser Beispiel = nas.zuhause.lan

2.4 das Zertifikat erstellen

(mit unserer CSR, dem privaten Schlüssel der CA, dem CA-Zertifikat und der Konfigurationsdatei)

Kommando: "openssl x509 -req -in <subdomain.SLD.TLD>.csr -CA <SLD>.pem -CAkey <SLD>.key -CAcreateserial -out <subdomain.SLD.TLD>.crt -days 3650 -sha256 -extfile <subdomain.SLD.TLD>.ext"

- unser Beispiel = SLD: zuhause - subdomain.sld.TLD: nas.zuhause.lan

> hey, ihr "Durchhalter...", ihr habt es tatsächlich geschafft... na, ja - fast! □□ Das Stammzertifikat muss ja noch verteilt werden, damit die Browser euer Zertifikat erfolgreich validieren können und der Web Server braucht ja noch die Information welches Zertifikat er verwenden soll, aber dann, versprochen, "klappt's auch mit dem Nachbarn" □□

Revision #11
Created 16 June 2025 16:02:05 by tomek
Updated 17 June 2025 10:29:10 by tomek