

Kostenloses SSL-Zertifikat

Wir empfehlen Deinen Downloadserver über das HTTPS Protokoll verfügbar zu machen. Ein entsprechendes Zertifikat kannst Du kostenlos von <https://letsencrypt.org/> erhalten. Um den Prozess der Zertifikatserstellung zu automatisieren, verwenden wir die Software <https://github.com/svrco/getssl>. getssl kann z. B. auch auf Deinem LoxBerry installiert werden, wenn Du keinen Server mit Root-Access zur Verfügung hast.

Getssl einrichten

Getssl ist ein reines Bash-Skript und läuft auf so ziemlich jedem Unixartigen Betriebssystem. Wenn Du es auf dem LoxBerry installieren möchtest, empfiehlt sich ein eigenes Unterverzeichnis in `/opt/loxberry/webfrontend/legacy`, also z.B. `/opt/loxberry/webfrontend/legacy/getssl`. Dieses Verzeichnis ist für eigene Skripte auf dem LoxBerry gedacht und wird auch nicht bei Updates überschrieben oder gelöscht.

Anschließend richtest Du für Deine Domain mit folgendem Befehl eine Grundkonfiguration ein:

```
getssl -c DEINEDOMAIN.tld
```

Hierbei wird ein neues Verzeichnis `DEINEDOMAIN.tld` erzeugt, in dem Du die Konfigurationsdatei `getssl.cfg` nun an Deine Gegebenheiten anpassen musst. Die [Dokumentation von getssl](#) sollte Dir hier weiterhelfen.

Unser Setup

Wir haben lediglich einen Webserver, der nicht per Rootshell erreichbar ist. Unser getssl läuft also auf

einem LoxBerry bei uns im Smarthome  Die Authentifizierung wird mit einem speziellen File auf dem Webserver durchgeführt, bei getssl heisst das: "Placing a file on the webserver containing the challenge token". Dieses File lassen wir per FTP direkt von getssl automatisch hochladen.

Eine weitere Schwierigkeit in unserem Setup ist, dass wir die erzeugten Zertifikate im Webfrontend unseres Webhosters jedes Mal manuell hochladen müssen. Wir lassen uns somit neu erzeugte Zertifikate automatisch per Email zusenden, sodass dann nur noch ein paar Klicks notwendig sind, um die Zertifikate zu aktivieren.

Getssl automatisieren

Wenn alles läuft, kannst Du getssl nun über einen Cronjob automatisieren. getssl erkennt dabei automatisch, ob deine Zertifikate noch gültig sind oder neu erzeugt werden müssen.

Mit folgendem Befehl werden die Zertifikate aller Deiner Domains geprüft und ggf. neu erzeugt:

```
getssl -w /opt/loxberry/webfrontend/legacy/getssl -a
```

Möchtest Du nur die Zertifikate einer einzelnen Domain prüfen und ggf. neu erzeugen, geht das mit:

```
getssl -w /opt/loxberry/webfrontend/legacy/getssl DEINEDOMAIN.tld
```

Beachte, dass wir hier immer die Working-Directory manuell setzen, sonst nutzt getssl standardmäßig `.getssl` im Homeverzeichnis des aktuellen Users.

Ein entsprechender Eintrag in Deiner crontab könnte also folgendermaßen aussehen, um getssl täglich um 3:00 Uhr aufzurufen:

```
SHELL=/bin/bash
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=""
#
# m h dom mon dow user  command
0 3 * * * loxberry /opt/loxberry/webfrontend/legacy/getssl/getssl -w
/opt/loxberry/webfrontend/legacy/getssl -a >/dev/null 2>&1
```

Spezialfall: Zertifikate per Mail senden

Wie erwähnt ist bei unserem Server ein weiterer Schritt notwendig, das wir die Zertifikate nicht automatisch von getssl auf dem Server speichern lassen können. Wir verschicken die neu erstellten Zertifikate an eine unserer Emailadressen, um sie von dort aus manuell auf den Server hochzuladen. Dazu verwenden wir folgendes Skript:

[cronjob.sh](#)

```
#!/bin/bash

TO="mail@example.com"

cd /opt/loxberry/webfrontend/legacy/getssl/getssl

for DOMAIN in * ; do
    if [ -d "$DOMAIN" ]; then
        echo "Checking $DOMAIN..."
        OUTPUT=`/opt/loxberry/webfrontend/legacy/getssl/getssl/getssl -w
/opt/loxberry/webfrontend/legacy/getssl $DOMAIN`
        echo $OUTPUT | grep -q 'is valid for more than';
        if [ $? -eq 1 ]; then
            echo "Cert for $DOMAIN was renewed. Sending by
email"
            echo "New certificates for $DOMAIN were
created. Please update." | \
                s-nail -a ".$DOMAIN/fullchain.crt" -a
"./$DOMAIN/$DOMAIN.crt" -a ".$DOMAIN/$DOMAIN.key" -s "New certificates
for $DOMAIN" $TO
        fi
    fi
done
```

done fi

Das Skript muss als `/opt/loxberry/webfrontend/legacy/getssl/getssl/cronjob.sh` gespeichert werden und wird dann einmal täglich aufgerufen. Neue Zertifikate werden dann an die Emailadresse unter `T0=` gesendet, sofern sie neu erzeugt worden sind.

Auf Debian-Systemen benötigt ihr noch das Paket `s-nail`, welches ihr wie folgt nachinstallieren könnt:

```
sudo apt-get install s-nail
```

Der Eintrag in der crontab sieht analog zum vorherigen Kapitel aus, nur dass nicht `getssl` direkt aufgerufen wird sondern unser `crontab.sh` Skript.

From:

<https://wiki.loxberry.de/> - **LoxBerry Wiki - BEYOND THE LIMITS**

Permanent link:

https://wiki.loxberry.de/organisatorisches/archiv/download_und_spiegelsites/kostenloses_ssl_zertifikat

Last update: **2024/04/02 06:28**