Voraussetzung: Debian 11 Server mit shh Verbindung, installiertem docker.io, docker-compose und OTRS.

1. Daten des OTRS löschen

systemctl stop otrs.service

rm -rf /srv/docker/otrs/data/\*

1. Verzeichnis erstellen

mkdir -p /srv/docker/nginx-proxy/ssl

mkdir -p /srv/docker/nginx-proxy/conf.d

cd /srv/docker/nginx-proxy/ssl

1. Erstellung des privaten Schlüssels

openssl genrsa -out proxy.key 4096

1. Erstellung des Certificate Signing Request

openssl req -new -key proxy.key -out proxy.csr

1. Die folgenden Daten eingeben

Proxy.csr

Country Name (2 letter code) [AU]:DE

State or Province Name (full name) [Some-State]:BE

Locality Name (eg, city) []:Berlin

Organization Name (eg, company) [Internet Widgits Pty Ltd]:JIKU

Organizational Unit Name (eg, section) []:

Common Name (e.g. server FQDN or YOUR name) []:debian

Email Address []:

Please enter the following 'extra' attributes to be sent with your certificate request

A challenge password []:

An optional company name []:

1. Erstellen eines Self-Sign-Certificate

openssl x509 -req -days 365 -in proxy.csr -signkey proxy.key -out proxy.crt

1. Wechsel in das Verzeichnis nach /srv/docker/nginx-proxy
2. Dann erstelle die Datei docker-compose.yml und trage folgendes ein und beachte unbedingt, YAML-Dateien benötigen eine bestimmte Formatierung. Die Einrückungen müssen

mit 2 x **[ Leertaste ]** eingerückt werden. Wenn es später zu Fehlermeldungen

während des Compose kommt, sind die Einrückungen nicht korrekt oder es fehlen bestimmte Leertasten zwischen dem **[ : ]** und Übergabewert:

docker-compose.yml

version: '2'

volumes:

docker.sock:

ssl:

conf.d:

services:

nginx-proxy:

image: jwilder/nginx-proxy

container\_name: nginx-proxy

ports:

- "443:443"

volumes:

- /var/run/docker.sock:/tmp/docker.sock:ro

- ./ssl:/etc/nginx/certs:ro

- ./conf.d:/etc/nginx/conf.d

whoami:

image: jwilder/whoami

container\_name: whoami

environment:

- VIRTUAL\_HOST=whoami.local

1. In das Verzeichnis /srv/docker/nginx-proxy/conf-d/ wechseln.
2. Die Datei otrs.conf erstellen und den Inhalt eintragen:

otrs.conf

upstream backend {

server IP-Adresse:80;

}

server {

listen 443 ssl

ssl\_certificate /etc/nginx/certs/proxy.crt;

ssl\_certificate\_key /etc/nginx/certs/proxy.key;

location / {

proxy\_pass http://IP-Adresse;

}

}

1. Wechsel in das Verzeichnis /srv/docker/nginx-proxy/ und starte den Container

docker-compose up -d

1. Erstelle die Datei start.sh

start.sh

#!/bin/sh

RC=0

cd /srv/docker/nginx-proxy

/usr/bin/docker-compose up -d || RC=1

exit $RC

1. Erstelle die Datei stop.sh

stop.sh

#!/bin/sh

RC=0

cd /srv/docker/nginx-proxy

/usr/bin/docker-compose down || RC=1

exit $RC

1. Berechtigungen ändern

chmod 755 start.sh

chmod 755 stop.sh

1. Erstelle die Datei /etc/systemd/system/nginx-proxy.service und trage folgendes ein:

nginx-proxy.service

[Unit]

Description=Docker - Nginx-Proxy Container

Requires=docker.service

After=docker.service

[Service]

Type=oneshot

RemainAfterExit=yes

ExecStart=/srv/docker/nginx-proxy/start.sh

ExecStop=/srv/docker/nginx-proxy/stop.sh

[Install]

WantedBy=default.target

1. Service aktivieren

systemctl enable nginx-proxy.service

1. Service starten

systemctl start nginx-proxy.service

1. Server neu starten
2. Den Browser starten und die URL <https://deine_ip-adresse> eingeben
3. Es kann sein das die Webseite beim ersten starten einen Bad Error ausgibt. Dann einfach ein paar Minuten warten. Meistens ist der Server noch nicht fertig.