LetsEncrypt, Powerdns manager und dns-01 challenge

Einrichtung:

Scripts:

```
#!/bin/bash
USER= #description from pdns panel
ID= #What pdns admin  says
PASS= # you know that one ;)

# we get: CERTBOT_DOMAIN
# CERTBOT_VALIDATION (key)
# CERTBOT_TOKEN (http only)
# CERTBOT_CERT_PATH (bla, same as CERTBOT_KEY_PATH and CERTBOT_SNI_DOMAIN

CREATE_DOMAIN="_acme-challenge.$CERTBOT_DOMAIN"

# alt: ID from pdns - -d id=$id \
curl -X GET -G https://$PATH/api/remote.php \
    -d action=updateRecord \
    -d desc=$USER \
    -d password=$PASS \
    -d domain=$CREATE_DOMAIN \
    -d content=$CERTBOT_VALIDATION \
    --trace-ascii -

COUNT=20
NEED=3
while [ $NEED -gt 0 ];do
        COUNT=$(( $COUNT -1 ))
        if [ $COUNT -lt 1 ];then
                echo "CHALLENGE FAILED - DNS NOT UP2DATE" 1>&2
                exit 1
        fi;
    NEED=3
    for I in 1 2 3;do
        dig +short -t TXT $CREATE_DOMAIN @ns$I.$DOM | grep $CERTBOT_VALIDATION  >/dev/null
        [ $? -eq 0 ] && NEED=$(( $NEED - 1 ))
    done
    sleep 1;
done
```

Renewal file:

```
[renewalparams]
authenticator = manual
pref_challs = dns-01,
installer = None

manual_auth_hook = /usr/sbin/dns.sh
manual_public_ip_logging_ok = true
renew_hook = /usr/sbin/renewposthook.sh
#optional, zB zum Certs verteilen, Services neu starten etc
```

pdns admin:

In der Records-Liste ein _acme-challenge.www.example.com anlegen - Typ: TXT. Kann leer sein, aber ein Leerzeichen ist notwendig

Speichern und ganz rechts auf den  klicken; hier "Add Password", eine Beschreibung und ein Kennwort eingeben.

Die Beschreibung ist im ersten Script unter ID= einzutragen, das Paßwort unter.. Paßwort.

Der erste Certbot-Aufruf ist dann:

certbot -a manual --preferred-challenge=dns-01 certonly --manual-auth-hook=/usr/sbin/dns.sh --manual-public-ip-logging-ok --renew-hook=/usr/sbin/renewposthook.sh -d <domain name>

danach sollte das Renew-File geprüft werden.

getestet mit certbot Version 0.10