Coturn (STUN / TURN Server für Jitsi Meet)

Weil einige Firewalls auf der Clientseite verhindern, sich mit UDP Port 10000 zu verbinden, um WebRTC durchzuschleusen, sollte ein TURN Server aufgesetzt werden, über den dann der Verkehr geleitet wird. Wenn UDP Ports möglich sind, stellt der STUN Server eine Peer2Peer Verbinung her. Wenn nicht, wird der Traffic über Port 443 geleitet, was bei jedem restriktiven Netzwerk funktionieren sollte.

> ⚠ **Wichtig** ⚠ Der TURN Server kann nicht ohne weiteres über einen Reverse Proxy geleitet werden, weil er nicht auf dem HTTP Protokoll beruht, sondern auf TCP und/oder TLS. Es sollte also am besten ein eigener Server sein, auf dem der Port 443 freigegeben werden kann, ohne auf einen Reverse Proxy zu zeigen.

https://hub.docker.com/r/coturn/coturn

https://doganbros.com/blog/turn-server-setup-for-jitsi-on-ubuntu-20-04-tls

jits meet .env

```
JVB_STUN_SERVERS=turn.DOMAIN.de:443

TURN_CREDENTIALS=rqj........GhY
TURN_HOST=turn.DOMAIN.de
TURN_PORT=443
#TURNS_HOST=turn.DOMAIN.de
#TURNS_PORT=443

TURN_TRANSPORT=tcp
ENABLE_TURN=1
ENABLE_P2P=1
```

certbot installieren

```
sudo apt update
sudo apt install certbot
sudo certbot certonly --standalone --preferred-challenges http -d turn.DOMAIN.de
sudo ufw allow 443
```

coturn docker-compose.yml

```
services:
    coturn:
        network_mode: host
        #networks:
         # - jitsi_meet.jitsi
        container_name: coturn
        image: coturn/coturn
        restart: unless-stopped
        volumes:
          -
/etc/letsencrypt/live/turn.DOMAIN.de/fullchain.pem:/etc/letsencrypt/live/turn.DOMAIN.de/fullch
ain.pem
          -
/etc/letsencrypt/live/turn.DOMAIN.de/privkey.pem:/etc/letsencrypt/live/turn.DOMAIN.de/privkey.
pem
        tmpfs:
          - /var/lib/coturn
        #ports:
          #- 80:3478
          #- 80:3478/udp
          #- 443:5349
          #- 443:5349/udp
          #- 5349:5349
          #- 5349:5349/udp
          #- 3478:3478
          #- 3478:3478/udp
          #- 80:80
          #- 80:80/udp
```

```
          #- 443:443
          #- 443:443/udp
        command:
          - --log-file=stdout
          - --verbose
          - --cert=/etc/letsencrypt/live/turn.DOMAIN.de/fullchain.pem
          - --pkey=/etc/letsencrypt/live/turn.DOMAIN.de/privkey.pem
          - --min-port=49160
          - --max-port=49200
          - --listening-port=443
          #- --tls-listening-port=443
          - --fingerprint
          - --no-multicast-peers
          #- --no-udp-relay
          #- --no-udp
          #- --no-tcp-relay
          #- --no-tcp
          - --no-cli
          - --no-tlsv1
          - --no-tlsv1_1
          - --external-ip=116.203.93.143
          - --static-auth-secret=rqj...........[openssl rand -base64 32].............cGhY
          - --use-auth-secret
          - --realm=turn.DOMAIN.de

#networks:
#  jitsi_meet.jitsi:
#      name: jitsi_meet.jitsi
#      external: true
#      driver: bridge
```

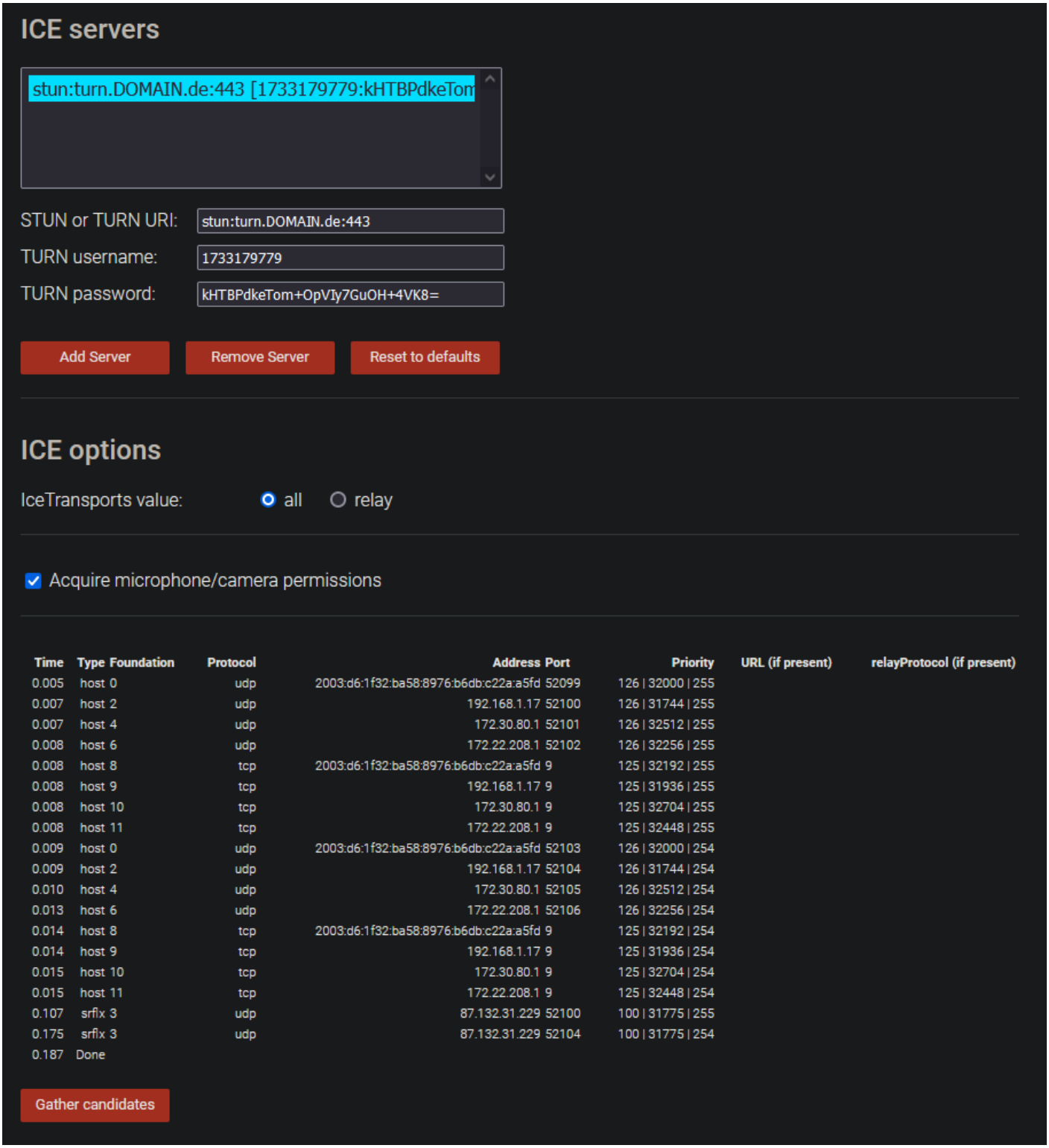
config testen:

```
secret=rqjw...........cGhY && time=$(date +%s) && expiry=8400 && username=$(( $time + $expiry )) &&echo username:$username && echo password : $(echo -n $username | openssl dgst -binary -sha1 -hmac $secret | openssl base64)
```

```
root@hetzner-02-ubnt-amd:~/jitsi# secret=rqjwJbHcp
st -binary -sha1 -hmac $secret | openssl base64)
username:1733113344
password : 5xjOaS/bTmRu/jt6a6W0mctC9L4=
```

und bei trickle-ice IM FIREFOX (chrome klappt nicht gut) angeben:

https://webrtc.github.io/samples/src/content/peerconnection/trickle-ice/



Revision #7
Created 2024-12-02 01:04:16 UTC
Updated 2024-12-02 20:44:03 UTC