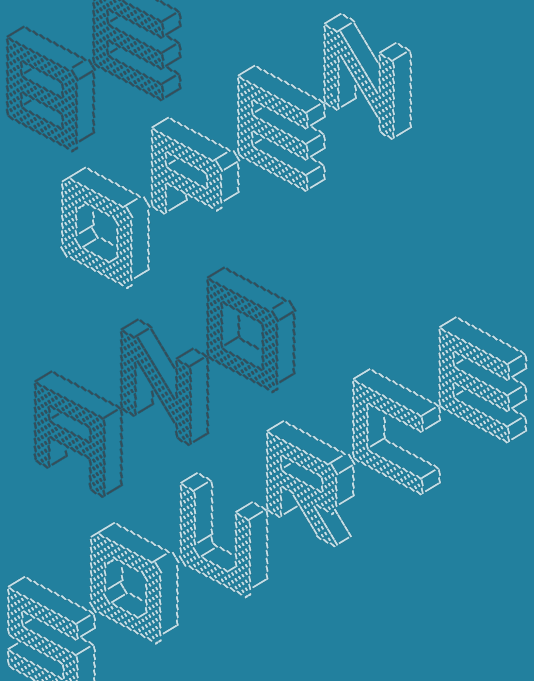
25. Mai 2023

OpenTalk Installations-Workshop

Stefan Sydow und Dennis Kalbhen



SLAC 2023
23.–25. Mai 2023 | Berlin

OpenTalk Installations-Workshop

Dennis Kabhen und Stefan Sydow [OpenTalk]

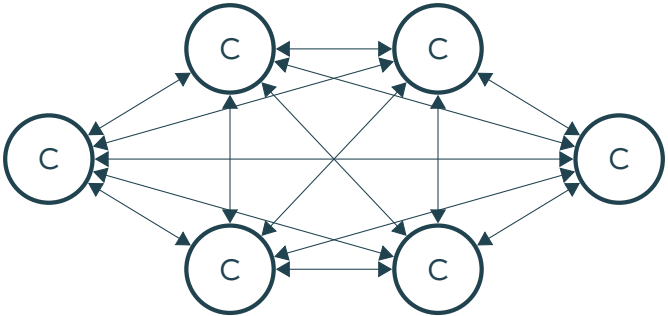
OpenTalk – Datenschutz und digitale Souveränität

- Ausgründung aus der Heinlein-Support GmbH
- Projekte: Freistaat Thüringen, Uni Greifswald, Bundesrat, UBA, uvm
- OT-Integration: UCS via Keycloak, Innovaphone myApps, mgm (A12), OX u. a.
- Open Source: `https://gitlab.opencode.de/opentalk/`

Was erwartet euch?

- Einführung
  - Überblick WebRTC
  - Service Architektur
- Praktischer Teil
  - Vorbereitungen
  - Repository + HowTo
  - OpenTalk Installation
  - API Beispiel
  - Skalierbarkeit + HA
- Weiterführendes
  - Staterecovery und Backup
  - Ansätze zur Fehlersuche

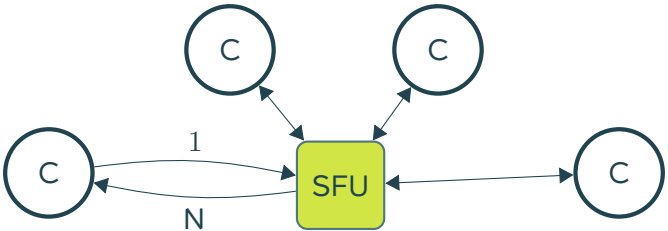
WebRTC ist doch P2P?!



Nachteile:

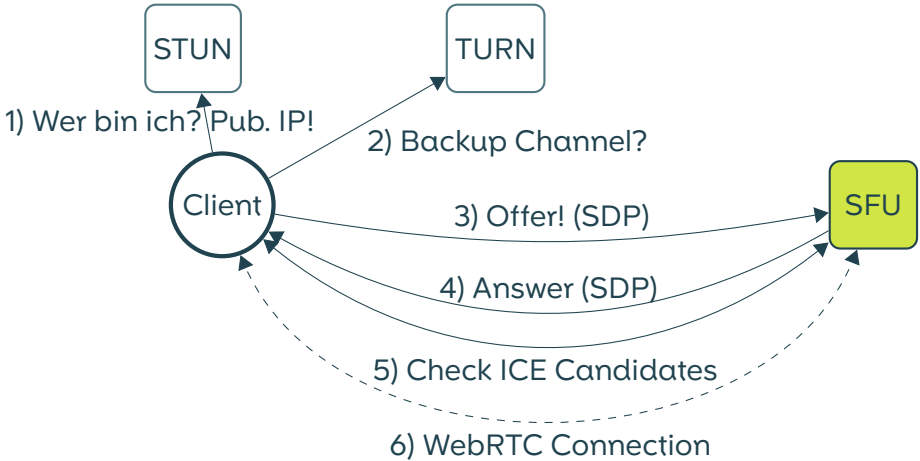
- Hoher Upstream am Client – senden an alle
- Backend hat keine dierekte Kontrolle
- Datenschutz – Teilnehmer IP-Adressen sichtbar

Selective Forwarding Unit (SFU)

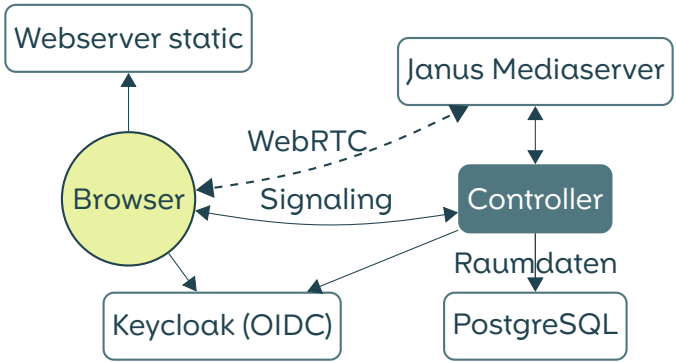


- Client: 1× senden / N× empfangen
- Mediaströme nur weiterleiten nicht umgerechnet
- dynamisch konfiguriebar

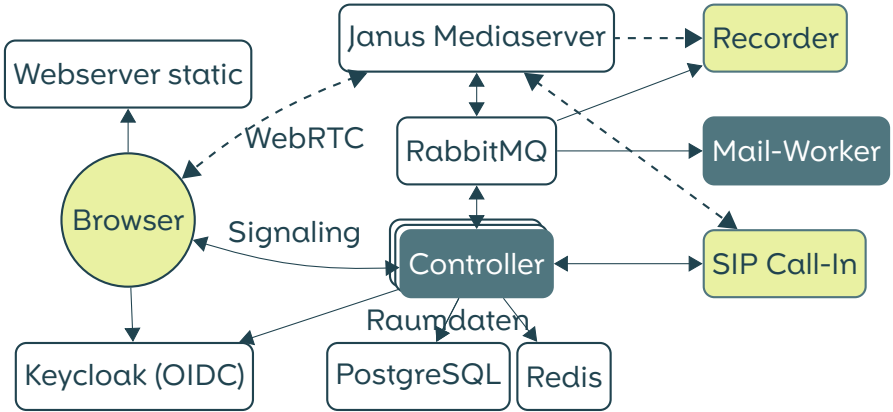
WebRTC Verbindungsaufbau (ICE)



Konferenz-Management



System Überblick



OpenTalk Installation

Vorab zum Praxisteil

- Was wir vorbereitet haben:
  - 30+ VMs
  - Debian 11
  - sshd
  - DNS
  - LE-Zerts und Webserver
- Wer will Installieren?
- Wer hat eine eigene VM?
  - Welches OS?
- Wer braucht einen Laptop für den ssh Zugriff?
- Was ist wenn wir aus der Zeit laufen?

Repository + HowTo

Code:

- `https://gitlab.opencode.de/opentalk/`

Install-Guide:

- `https://gitlab.opencode.de/opentalk/ot-setup`

Alles bereit?

- Server Ressourcen: `lscpu`, `free -h`, `df -h`, `top`, `htop`

Alles bereit?

- Server Ressourcen: `lscpu, free -h, df -h, top, htop`
- Docker Engine: d`ocker -v; docker compose version`

Alles bereit?

- Server Ressourcen: `lscpu, free -h, df -h, top, htop`
- Docker Engine: d`ocker -v; docker compose version`
- FW Ports: `nftables, iptables, telnet`

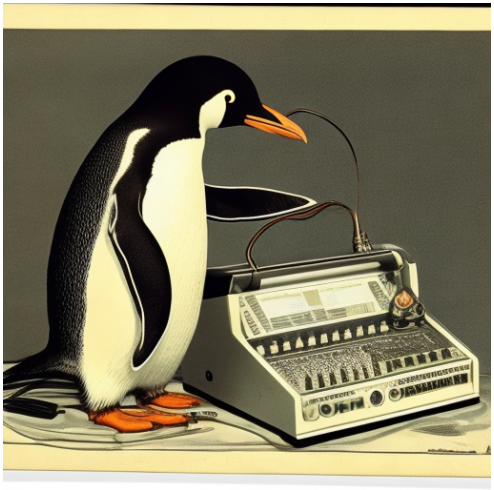
Alles bereit?

- Server Ressourcen: `lscpu, free -h, df -h, top, htop`
- Docker Engine: d`ocker -v; docker compose version`
- FW Ports: `nftables, iptables, telnet`
- DNS Records: `drill, dig`

Alles bereit?

- Server Ressourcen: `lscpu, free -h, df -h, top, htop`
- Docker Engine: d`ocker -v; docker compose version`
- FW Ports: `nftables, iptables, telnet`
- DNS Records: `drill, dig`
- Webserver: `nginx -t, systemctl status, nginx, curl`, Browser
- Editor: vim, kenne keinen anderen Editor ☺

Los geht's!



Quick Question

Was kostet mehr Bandbreite?

A) Viele Räume mit wenigen Nutzern (100 Räume, 9 Nutzer je Raum).
B) Wenige Räume mit richtig vielen Nutzern (3 Räume, 300 Nutzer je Raum).

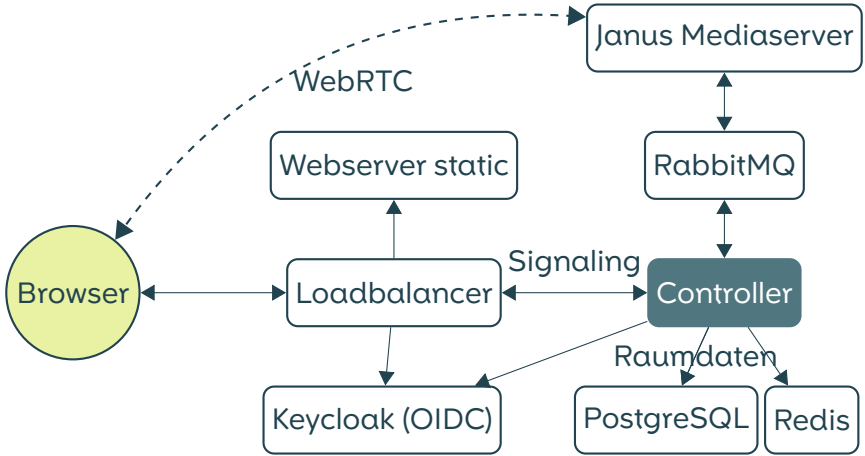
API Beispiel

Keycloak Konfiganpassung:

- https://accounts.slacX.opentalk.run/auth
- Enable - Direct access grants

```
#Get ACCESS TOKEN
TOKEN=$(curl -s -d 'client_id='${OT_CLIENT_ID}'' \
                -d 'username='${ACCOUNT_USERNAME}'' \
                -d 'password='${ACCOUNT_PASSWORD}'' \
                -d 'scope=openid' \
                -d 'grant_type=password' \
                  'https://'${OT_KEYCLOAK_URL}'/auth/realms/'${OT_REALM_NAME}'/
    protocol/openid-connect/token' | jq -r .access_token)
```

Skalierbarkeit und Fail-Over



Anhang

Der WebRTC Standard

- Peer-to-Peer Media-Streaming im Browser
  – geringe Einstiegshürde
- Geschichte:

  2010 → Google
  2011 IETF Working Group
  2021 Spezifikation: WebRTC 1.0
- Umfang: Browser-API, Peer-to-Peer, gängige Protokolle (Kombination aus SRTP, DTLS, SIP . . .)

WebRTC Browser Implementierung

– weit fortgeschritten aber nicht abgeschlossen[1,2]

- stabil: Verbindungsaufbau, Audio- und Videoübertragung
- beta: Diagnose- und Statistikfunktionen – Browser-spezifisch
- experimentell: SVC-Codecs
- Browser-Familien:
  - Chrome: Vorreiter (entwickeln libwebrtc)
  - Firefox: Nutzt libwebrtc
  - Safari: Plattformfokus auf nativen Apps

[1] Stand Nov. 2020: `https://w3c.github.io/webrtc-interop-reports/webrtc-pc-report.html`
[2] Stand Heute: `https://wpt.fyi/results/?label=master&label=experimental&aligned&q=rtc`