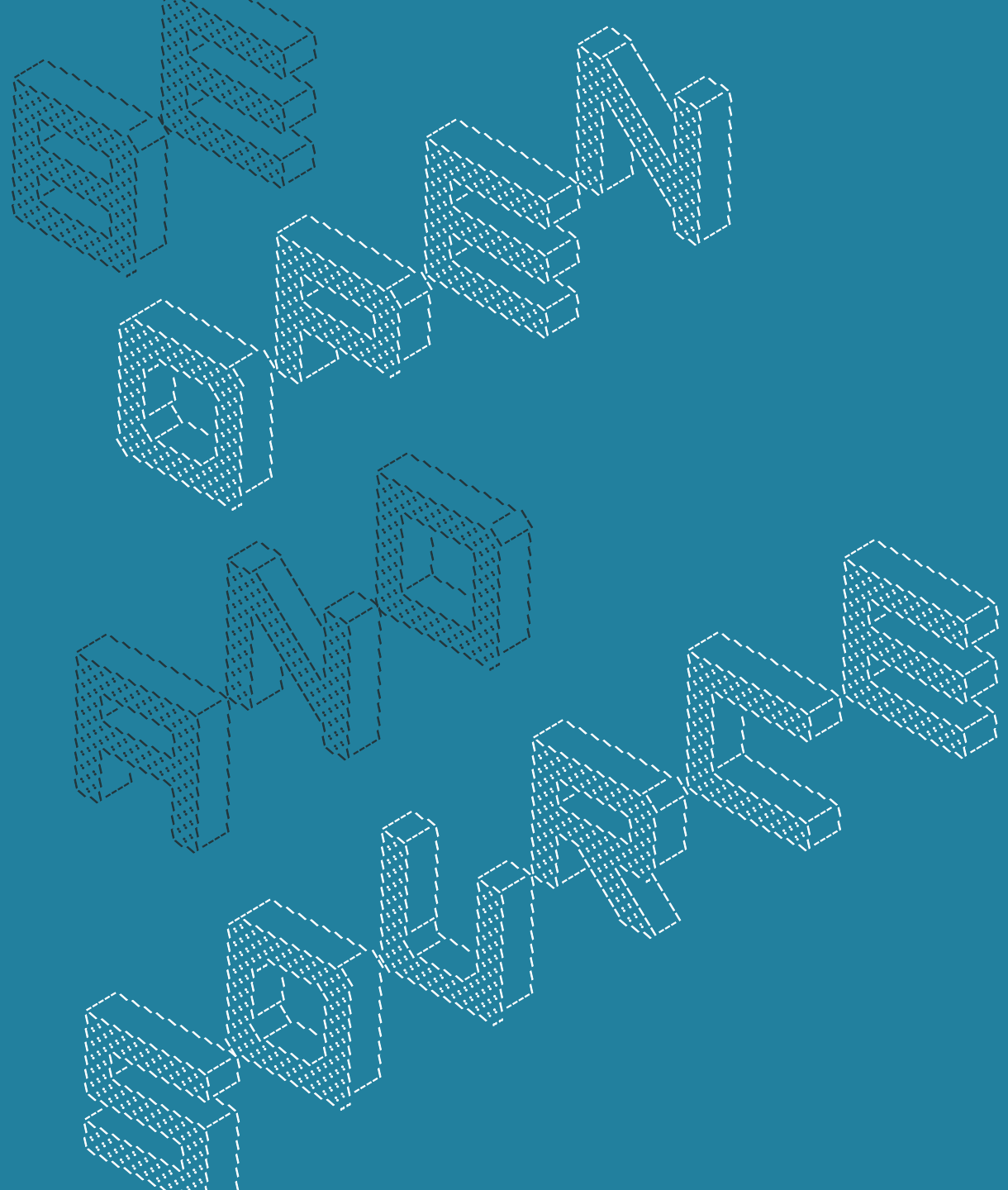


SLAC 2024

06.-08. Mai 2024 | Berlin

www.slac-2024.de

OpenTalk
Installationsworkshop

Wir haben Zeit!

Wir probieren aus.

Wir arbeiten zusammen.

Fragen? Jederzeit bitte!

OpenTalk
Erfahrungen?

Agenda

- Betriebsszenarien
- Installation
  - default vs angepasst
- TURN

OpenTalk
Betriebsszenarien

OpenTalk Betriebsszenarien

- Single Host
  - alle Komponenten, ein Host

- Multi Host
  - OT Core + dedizierte(r) Janus Host(s)

    oder
  - alles verteilt, ausfallsicher und skaliert

OpenTalk Betriebsszenarien

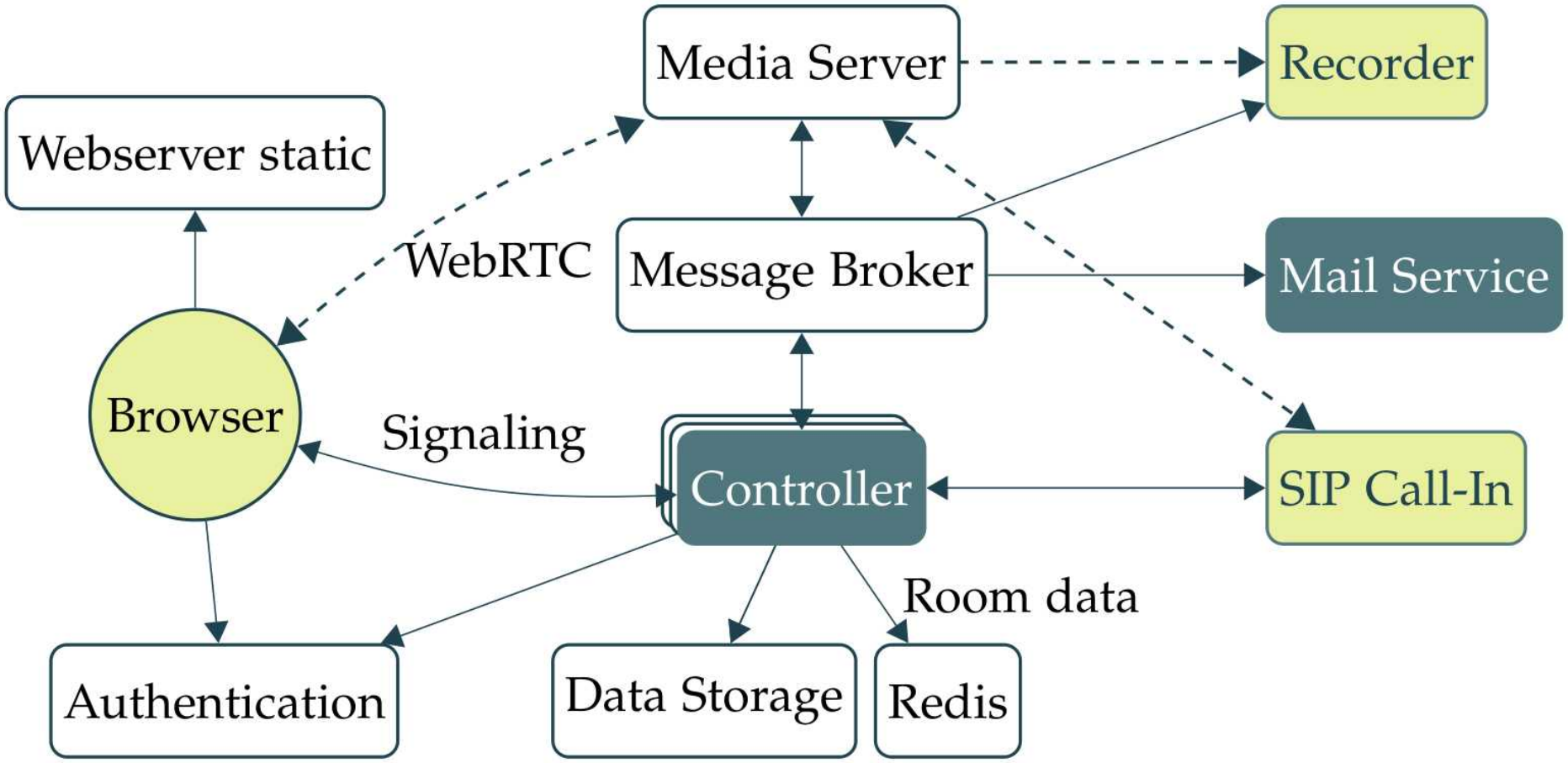
- Multi Host: alles verteilt, ausfallsicher und skaliert
- **Wirklich?**

- komplexe Infrastruktur

  vs

- so simpel wie möglich

Single Host


Media Server
Recorder
Webserver static
WebRTC
Message Broker
Mail Service
Browser
Signaling
Controller
SIP Call-In
Room data
Authentication
Data Storage
Redis

Praxis:

Single Host Installation

Single Host Installation

- Eure Workshop-Systeme:
  - workshop1.opentalk.run
  - workshop2.opentalk.run
  - workshop3.opentalk.run
  - ...
- SSH-Login siehe Handout
- https://gitlab.opencode.de/opentalk/ot-setup

Single Host Installation

- erstmal umschauen
  - **docker ps -a**

  - **cd /opt/opentalk**
  - **docker compose down**

Single Host Installation

- neue Installation
  - **mkdir /opt/workshop**
  - **git clone** https://gitlab.opencode.de/opentalk/ot-setup.git **/opt/workshop**

  - **cd /opt/workshop**
  - **git checkout v24.4.1**

Single Host Installation

- Konfig-Dateien erzeugen
  - **wir stehen in /opt/workshop**
  - **cp env.sample .env**
  - **cp extras/opentalk-samples/controller.toml.sample config/controller.toml**

Single Host Installation

- Anpassung Umgebungsvariablen
  - bash extras/gen-secrets.sh
  - Secrets/Output in die Zwischenablage

  - vim .env
    - OT_DOMAIN="workshopX.opentalk.run"
    - Secrets aus Zwischenablage einfügen

Single Host Installation

- Anpassung der Controller Konfiguration 1/2
  - **Secrets übertragen**
  - **Befehle aus der Installationsanleitung nutzen**

    siehe https://gitlab.opencode.de/opentalk/ot-setup#add-the-secretes-to-the-configcontrollertoml

    source .env; sed -i "s/postgrespw/$POSTGRES_PASSWORD/g" config/controller.toml

    source .env; sed -i "s/keycloakclientsecretforcontroller/$KEYCLOAK_CLIENT_SECRET_CONTROLLER/g" config/controller.toml

    source .env; sed -i "s/spacedeckapitoken/$SPACEDECK_API_TOKEN/g" config/controller.toml

    source .env; sed -i "s/etherpadapikey/$ETHERPAD_API_KEY/g" config/controller.toml

Single Host Installation

- Anpassung der Controller Konfiguration 2/2
  - **vim config/controller.toml**
  - **[http]**
    cors.allowed_origin = ["https://workshopX.opentalk.run"]
  - **[keycloak]**
    # URL to the keycloak
    base_url = "https://accounts-workshopX.opentalk.run/auth"

Single Host Installation

- Infrastrukturabhängige Anpassungen

- Service URLs
  - **service.base.domain.de**

    vs
  - **service-base.domain.de**
- Firewall UDP-Portrange

Single Host Installation

- Defaults der Umgebungsvariablen überschreiben
  - vim .env
  - KC_HOSTNAME="accounts-$OT_DOMAIN"
  - CONTROLLER_DOMAIN="controller-$OT_DOMAIN"
  - OPENTALK_CTRL_KEYCLOAK__BASE_URL="https://accounts-$OT_DOMAIN/auth"
  - SD_ENDPOINT="https://whiteboard-$OT_DOMAIN"
  - EP_ENDPOINT="https://pad-$OT_DOMAIN"
  - JANUS_UDP_PORT_RANGE="20000-25000"

Single Host Installation

Läufts?

- docker compose up -d
- docker compose logs -f keycloak

- im Browser: **https://workshopX.opentalk.run/**
- User: **testuser**
- Passwort: **!qayse45**

Single Host Installation

Testuser aktivieren in Keycloak

- im Browser: https://accounts-workshopX.opentalk.run/auth/

- User: admin
- Passwort?

Hört und sieht man mich?

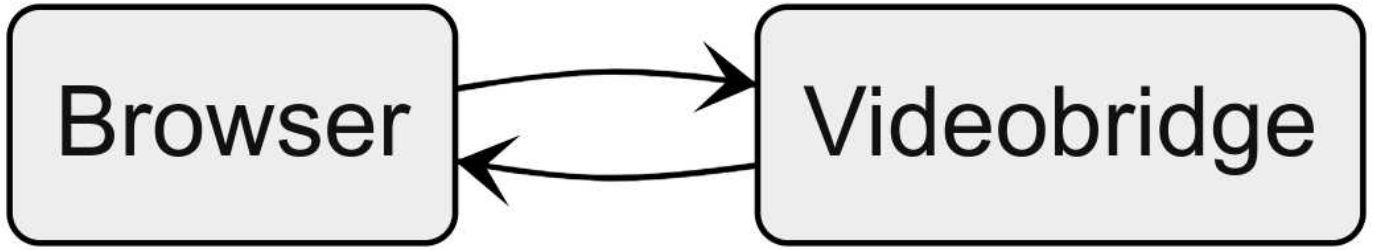
Meeting erstellen und testen

OpenTalk TURN

TURN Server

- Proxykompatibilität
- TCP 443
- wird via Controller/Signaling announced
- dedizierte Ressource
- muss im Controller aktiviert werden

TURN Server



Praxis:

TURN Installation & Konfiguration

TURN Server (coturn)

- /etc/turnserver.conf

```
tls-listening-port=443
listening-ip=128.140.34.146
fingerprint
use-auth-secret
static-auth-secret=janitor-dance-swimming
realm=turn.opentalk.run
cert=/etc/letsencrypt/live/turn.opentalk.run/fullchain.pem
pkey=/etc/letsencrypt/live/turn.opentalk.run/privkey.pem
syslog
no-multicast-peers
denied-peer-ip=0.0.0.0-0.255.255.255
...
```

TURN Server

- vim config/controller.toml

```
[turn]
lifetime = 86400

[[turn.servers]]
uris = [
    "turn:turn.opentalk.run:443?transport=udp",
    "turn:turn.opentalk.run:443?transport=tcp",
    "turns:turn.opentalk.run:443?transport=udp",
    "turns:turn.opentalk.run:443?transport=tcp",
]
pre_shared_key = "janitor-dance-swimming"

[stun]
uris = ["stun:turn.opentalk.run:3478"]
```

TURN Server

- vim config/controller.toml

```
[turn]
lifetime = 86400

[[turn.servers]]
uris = [
    "turn:turn.opentalk.run:443?transport=udp",
    "turn:turn.opentalk.run:443?transport=tcp",
    "turns:turn.opentalk.run:443?transport=udp",
    "turns:turn.opentalk.run:443?transport=tcp",
]
pre_shared_key = "janitor-dance-swimming"

[stun]
uris = ["stun:turn.opentalk.run:3478"]
```

- docker compose up -d --force-recreate controller

TURN Test!

Meeting erstellen und testen

via Proxy oder lokalen Firewall Limitierungen

OpenTalk

Fragen, Wünsche Anregungen

OpenTalk
Vielen Dank!