

# Installation von „Jitsi Meet“: Videokonferenzsystem aus Docker Containern

Michael Dörr, Sa. 18.04.2020

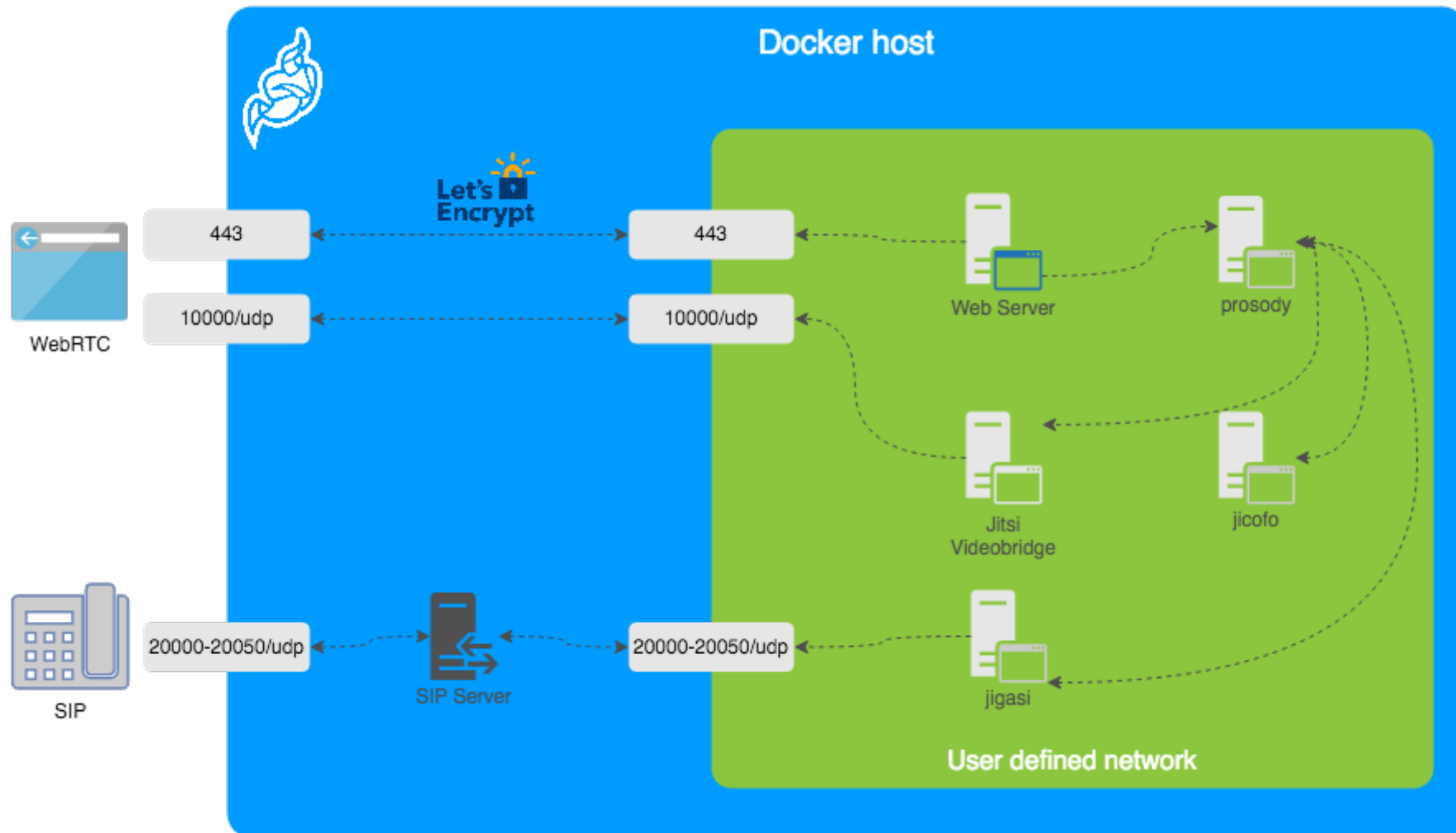
# Agenda

1. Motivation
2. Installation Virtualisierungsplattform Proxmox VE 6.1
3. Installation VM mit Ubuntu-Server 18.04
4. Installation „Jitsi Meet“ als Container
5. Praktische Erfahrungen
6. Ausblick: Single-Node Kubernetes-Cluster
  
7. Demo

# Blick hinter die Kulissen von „Jitsi Meet“

- Viele Einzelkomponenten:
  - Web interface ([nginx](#))
  - XMPP server ([prosody](#))
  - [jicofo](#): **conference focus** component
  - [jvb](#): **video bridge** (>= 1 router)
  - [jigasi](#): SIP gateway (optional)
  - [jibri](#) : **broadcasting infrastructure** (optional): Videostream-Aufzeichnung
  - [etherpad](#): gemeinsames Bearbeiten von Dateien
- Installation von „Jitsi Meet“ kann mit dem Paketmanager von Ubuntu erfolgen:
  - <https://jitsi.org/tutorial>
  - <https://github.com/jitsi/jitsi-meet/blob/master/doc/quick-install.md>
- Oder mit Docker Containern, die eine Abschottung der einzelnen Anwendungsteile ermöglichen:
  - <https://github.com/jitsi/docker-jitsi-meet>

# Architekturbild der „Jitsi Meet“ Container



# Vorbereitung für „Docker“

- Software-Pakete für `git` und `docker-compose` installieren:

- `apt install git`

- `apt install docker-compose`

- Reboot

```
root@jitsi2:~# apt search docker-compose
Sorting... Done
Full Text Search... Done
docker-compose/bionic,bionic 1.17.1-2 all
  Punctual, lightweight development environments using Docker

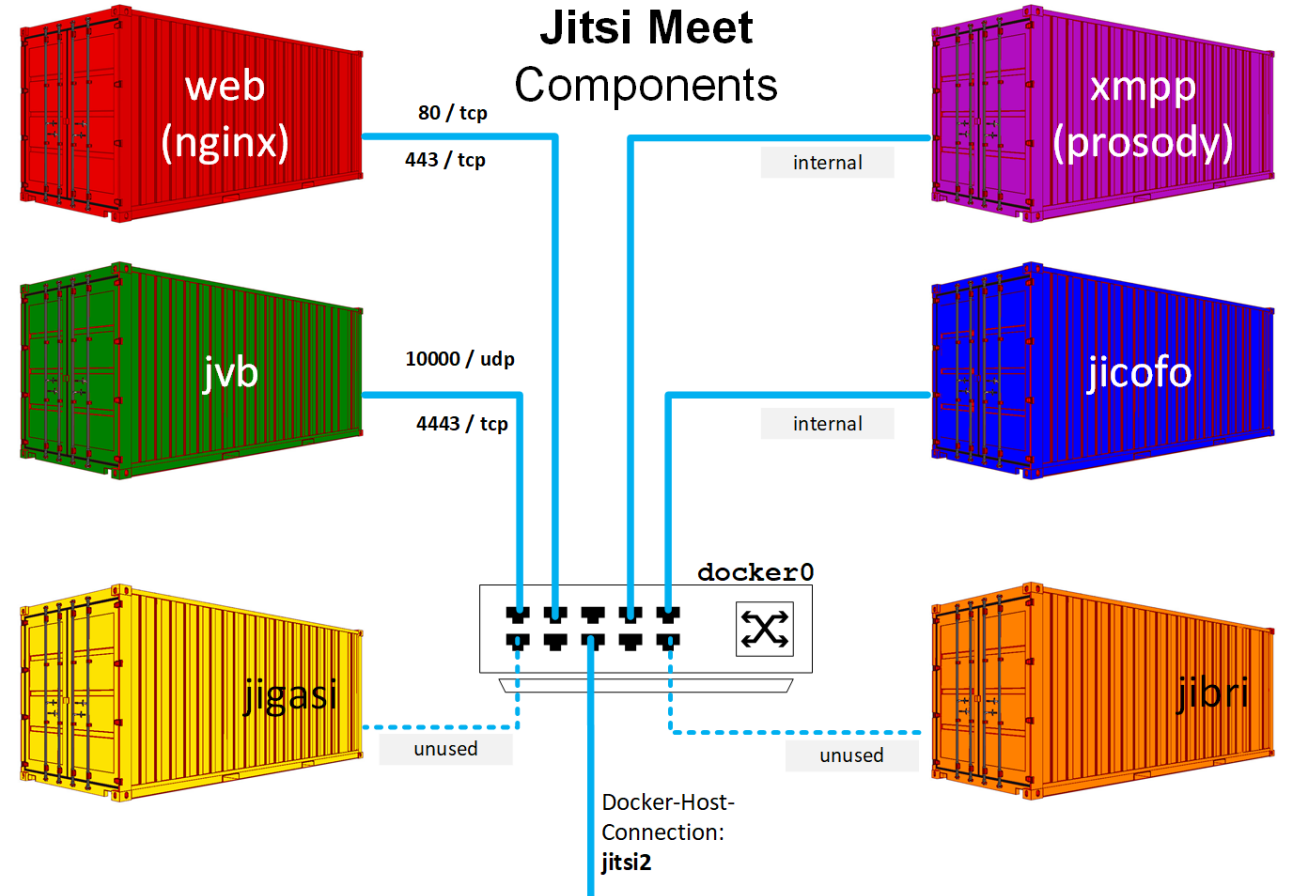
root@jitsi2:~# apt install docker-compose
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  bridge-utils cgroupfs-mount containerd docker.io golang-docker-credential-helpers
  libpython-stdlib libpython2.7-minimal libpython2.7-stdlib libsecret-1-0 libsecret-common pigz
  python python-asn1crypto python-backports.ssl-match-hostname python-cached-property
  python-certifi python-charset-normalizer python-chardet python-cryptography python-docker
  python-dockerpty python-dockerpycreds python-docopt python-enun34 python-funcsigs
  python-functools32 python-idna python-ipaddress python-jsonschema python-minimal python-mock
  python-openssl python-pbr python-pkg-resources python-requests python-six python-texttable
  python-urllib3 python-websocket python-yaml python2.7 python2.7-minimal runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools debootstrap docker-doc rinse zfs-fuse | zfsutils python-doc python-tk
  python-cryptography-doc python-cryptography-vectors python-enun34-doc python-funcsigs-doc
  python-mock-doc python-openssl-doc python-openssl-dbg python-setuptools python-socks python-ntlm
  python2.7-doc binutils binfmt-support
The following NEW packages will be installed:
  bridge-utils cgroupfs-mount containerd docker-compose docker.io golang-docker-credential-helpers
  libpython-stdlib libpython2.7-minimal libpython2.7-stdlib libsecret-1-0 libsecret-common pigz
  python python-asn1crypto python-backports.ssl-match-hostname python-cached-property
  python-certifi python-charset-normalizer python-chardet python-cryptography python-docker
  python-dockerpty python-dockerpycreds python-docopt python-enun34 python-funcsigs
  python-functools32 python-idna python-ipaddress python-jsonschema python-minimal python-mock
  python-openssl python-pbr python-pkg-resources python-requests python-six python-texttable
  python-urllib3 python-websocket python-yaml python2.7 python2.7-minimal runc ubuntu-fan
0 upgraded, 45 newly installed, 0 to remove and 0 not upgraded.
Need to get 69.8 MB of archives.
After this operation, 346 MB of additional disk space will be used.
Do you want to continue? [Y/n] _
```

# Anleitung von Golem.de (März 2020)

- Die Anleitung von **Golem.de** ist eigentlich nur eine ...
  - <https://www.golem.de/news/homeoffice-videokonferenzen-auf-eigenen-servern-mit-jitsi-meet-2003-147239.html>
- ... Übersetzung der Jitsi-Originalanleitung von Github:
  - <https://github.com/jitsi/docker-jitsi-meet/blob/master/README.md>
- Zuerst wird das Projektverzeichnis von Github geklont,
- dann wird eine Konfigurationsdatei (**.env**) angepasst,
- zuletzt werden mit dem Kommando **docker-compose**
  - die 4 Anwendungscontainer vom Docker-Hub heruntergeladen,
  - zusammen gestartet und über ein internes Netzwerk
  - und eine Docker-Bridge miteinander verbunden.

# Das ist (fast) alles:

- `git clone https://github.com/jitsi/docker-jitsi-meet`
- `cd docker-jitsi-meet`
- `cp env.example .env`
- `nano .env`
- `./gen-passwords.sh`
- `docker-compose up -d`



# Konfiguration für „jitsi1.agmc1.de“

```
QEMU (jitsi2) - noVNC - Mozilla Firefox
https://192.168.1.201:8006/?console=kvm&novnc=1&vmid=100&vmname=jitsi2&node=pve-002&resize=...
root@jitsi2:~/docker-jitsi-meet# ls -la
total 188
drwxr-xr-x 14 root root 4096 Apr 14 08:13 .
drwx----- 8 root root 4096 Apr 14 11:13 ..
drwxr-xr-x 3 root root 4096 Apr 13 22:14 base
drwxr-xr-x 2 root root 4096 Apr 13 22:14 base-java
-rw-r--r-- 1 root root 3307 Apr 13 22:14 CHANGELOG.md
-rw-r--r-- 1 root root 4655 Apr 13 22:14 docker-compose.yml
-rw-r--r-- 1 turtle turtle 8752 Apr 14 08:13 .env
-rw-r--r-- 1 root root 8413 Apr 13 22:14 env.example
-w-r--r-- 1 root root 8413 Apr 13 22:14 env.exmpl0
-w-r--r-- 1 turtle turtle 8750 Apr 13 16:56 env.msds5
-w-r--r-- 1 turtle turtle 8752 Apr 14 08:13 env.msds6
drwxr-xr-x 3 root root 4096 Apr 13 22:14 etherpad
-rw-r--r-- 1 root root 269 Apr 13 22:14 etherpad.yml
drwxr-xr-x 5 root root 4096 Apr 13 22:14 examples
-rwxr-xr-x 1 root root 870 Apr 13 22:14 gen-passwords.sh
drwxr-xr-x 8 root root 4096 Apr 13 22:14 .git
-rw-r--r-- 1 root root 48 Apr 13 22:14 .gitignore
drwxr-xr-x 3 root root 4096 Apr 13 22:14 jibri
-rw-r--r-- 1 root root 928 Apr 13 22:14 jibri.yml
drwxr-xr-x 3 root root 4096 Apr 13 22:14 jicofo
drwxr-xr-x 3 root root 4096 Apr 13 22:14 jigasi
-rw-r--r-- 1 root root 1475 Apr 13 22:14 jigasi.yml
drwxr-xr-x 3 root root 4096 Apr 13 22:14 jvb
-rw-r--r-- 1 root root 11359 Apr 13 22:14 LICENSE
-rw-r--r-- 1 root root 1015 Apr 13 22:14 Makefile
drwxr-xr-x 3 root root 4096 Apr 13 22:14 prosody
-rw-r--r-- 1 root root 21280 Apr 13 22:14 README.md
drwxr-xr-x 2 root root 4096 Apr 13 22:14 resources
drwxr-xr-x 3 root root 4096 Apr 13 22:14 web
root@jitsi2:~/docker-jitsi-meet#
```

```
QEMU (jitsi2) - noVNC - Mozilla Firefox
https://192.168.1.201:8006/?console=kvm&novnc=1&vmid=100&vmname=jitsi2&node=pve-002&resize=...
26c26
< JIBRI_XMPP_PASSWORD=
---
> JIBRI_XMPP_PASSWORD=956467fe76c5c817c91d9e7f0c4bce0a
37c37
< HTTP_PORT=8000
---
> HTTP_PORT=80
40c40
< HTTPS_PORT=8443
---
> HTTPS_PORT=443
43c43,44
< TZ=Europe/Amsterdam
---
> #TZ=Europe/Amsterdam
> TZ=Etc/UTC
46a48
> PUBLIC_URL=https://jitsi1.agmc1.de
50c52,53
< #DOCKER_HOST_ADDRESS=192.168.1.1
---
> DOCKER_HOST_ADDRESS=192.168.1.200
< #DOCKER_HOST_ADDRESS=172.23.42.88
58c61
< #ENABLE_LETSENCRYPT=1
---
> ENABLE_LETSENCRYPT=1
61a65
> LETSENCRYPT_DOMAIN=jitsi1.agmc1.de
64a69
> LETSENCRYPT_EMAIL=pa[redacted]@ail.com
319c324
< #ENABLE_HTTP_REDIRECT=1
---
> ENABLE_HTTP_REDIRECT=1
root@jitsi2:~/docker-jitsi-meet# diff env.example .env _
```



# Was fehlt noch?

- Besorgung einer „kurzen“ DNS-Adresse:
  - Angenehm für die Anwender
  - FQDN ist wichtig für Zertifikate von Let's-Encrypt (https://)
- Firewall-Freigaben im Betriebssystem
- Port-Forwarding in der FritzBox

- Ubuntu uncomplicated firewall:

- `ufw enable`
- `ufw allow ssh`
- `ufw allow 80/tcp`
- `ufw allow 443/tcp`
- `ufw allow 4443/tcp`
- `ufw allow 10000/udp`
- `ufw status`

Freigaben

Status	Bezeichnung	Protokoll	IP-Adresse im Internet	Port extern vergeben
●	HTTP-Server	TCP	79.249.198.140	80
●	HTTPS-Server	TCP	79.249.198.140	443
●	RTP	TCP	79.249.198.140	4443
●	RTP	UDP	79.249.198.140	10000

Jitsi Meet

jitsi1.agmc1.de

jitsi.org

# Secure, fully featured, and completely free video conferencing

Go ahead, video chat with the whole team. In fact, invite everyone you know. Jitsi Meet is a fully encrypted, 100% open source video conferencing solution that you can use all day, every day, for free — with no account needed.

**Start a new meeting**

ObnoxiousDragonsDevourStlyty **GO**

Mar 26th, 2020 4:43 PM

LinuxTag-2020 20:59

jitsi.org

Linux Tag 2020 4

Mute / Unmute

jitsi.org

Linux Tag 2020 1

jitsi.org

Linux Tag 2020 2

Connection: Good

- Bitrate: 1 N/A | 3233 Kbps
- Packet loss: 1 0% | 0%
- Resolution: 1280x720
- Frame rate: 30
- Server count: 1

Show less

Estimated bandwidth: 1 N/A | 5388 Kbps

- Remote address: 172.23.42.88
- Remote port: 10000
- Local address: 172.23.42.243
- Local port: 63578
- Transport: udp

Mimi

Mimi is having connectivity issues...