

# Prosody XMPP Server

Run your own XMPP Server

Emanuel Duss

2015-10-04

- Originally named Jabber (1998)
- Communication Protocol based on XML
- Features
  - Instant Messaging
  - Presence Information
  - Contact Lists
  - VoIP, Video, File Transfer, IoT, Social Networks, ...
- Used by
  - Google Talk (2005 - 2014)
  - AIM (2008 - 2011)
  - Facebook Chat (2010 - 2014)

- Jabber
  - Protocol Developed in 1999
  - First Jabber Server in 1999
  - First IM Service “Jabber.org” in 1999, still active and free to use
- XMPP Standard
  - 2002: IETF XMPP Working Group
  - 8 RFCs for XMPP Specification
  - RFC 6120: XMPP: Core
  - RFC 6121: XMPP: Instant Messaging and Presence
- XMPP Extensions
  - XMPP Standards Foundation (XSF)
  - Originally called “Jabber Software Foundation”
  - XMPP Extension Protocols (XEPs)
  - Service Discovery, Multi-User Chat, Publish-Subscribe, File Transfer
  - Internet of Things (IoT): Sensor Data, Control, Discovery, ...

- Decentralization: No central servers like AIM, MSN
- Open standards
- Widely used, many implementations
- Security: Isolated servers (intranet), secure authentication, TLS
- Flexibility, many extensions
- But:
  - No QoS
  - Network overhead, because of speaky XML
  - Not designed for big file transfers
  - No End-to-End encryption (use OTR!)

## OTR provides

- End-to-End encryption
- Authentication
- Deniability
- Perfect Forward Secrecy

## Usage

- It's not an XMPP feature
- Your client has to support OTR (“inside XMPP”)

## More information

- Project page: <https://otr.cyberpunks.ca/>
- Protocol specification: <https://otr.cyberpunks.ca/Protocol-v3-4.0.0.html>

# Why running an on XMPP server?

- Free server since 2004: `jabber.ccc.de`
- I used it for years.
- It was not so stable all the time.
- Denial of Service attacks
- More and more users → decentralized?



CCC Jabber Service @jabbercccde - 29. Sep.

This server has been temporarily switched off intentionally. The reason is neither government interference nor legal threat (1/2)

Please stay calm, we won't delete your contact lists. Please use the time to setup/use new jabber servers out there. (2/2)

<https://twitter.com/jabbercccde/status/648871207962562560>

- Up again on 1. October 2015
- De facto central jabber service in CCC scene
- Should be more decentralized!
- New admin team.
- Request from CCC: Setup your own jabber server or use another server.
- Don't take it for granted. It's a voluntary service.
- Press release: <https://ccc.de/de/updates/2015/jabbercccde>



Run your own XMPP server!

- Website: <https://prosody.im/>
- Easy to set up and to configure
- Efficient with system resources
- Open Source (MIT/X11 License)
- Nice documentation.

## First impression

- First impression was good, so I gave it a try.
- I haven't tried other XMPP servers like ejabberd, so I can't compare them.

# Install Prosody

Add the repository for the latest version and the PGP key:

```
# cat /etc/apt/sources.list.d/prosody.list
deb http://packages.prosody.im/debian wheezy main

# wget https://prosody.im/files/prosody-debian-packages.key -O- \
  | sudo apt-key add -
```

Install Prosody:

```
# apt-get install prosody
```

LuaSec 0.5 is required for proper TLS support (certificate verification, Perfect Forward Secrecy, CRL). If 0.5 is not available in your distribution, install the fork `lua-sec-prosody`.

```
# apt-get install lua-sec-prosody
```

# Configure Prosody

Copy configuration:

```
# cp /etc/prosody/conf.avail/{example.com,jabber.motd.ch}.cfg.lua
```

Edit configuration:

```
# vi /etc/prosody/conf.avail/jabber.motd.ch.cfg.lua
VirtualHost "jabber.motd.ch"
enabled = true -- Remove this line to enable this host
ssl = {
    key = "/etc/prosody/certs/jabber.motd.ch.key";
    certificate = "/etc/prosody/certs/jabber.motd.ch.crt";
}
Component "conference.jabber.motd.ch" "muc"
```

Force TLS:

```
# vi /etc/prosody/prosody.cfg.lua  
[...]  
c2s_require_encryption = true  
s2s_require_encryption = true  
[...]
```

## Get your X.509 certs

Generate private key and certificate signing request:

```
$ CN=jabber.motd.ch
$ openssl req -newkey rsa:4096 -subj /CN="$CN" -nodes \
  -keyout "$CN.key" -out "$CN.csr"
$ ls $CN.{key,csr}
jabber.motd.ch.csr jabber.motd.ch.key
```

Now let's sign your CSR to get your `jabber.motd.ch.crt`. I used StartSSL, since Let's Encrypt is not yet available for public. Don't forget to provide a proper X.509 certificate chain if needed!

```
# chown prosody.prosody /etc/prosody/certs/jabber.motd.ch.{key,crt}
# curl https://startssl.com/certs/sub.class1.server.ca.pem \
  >> /etc/prosody/certs/jabber.motd.ch.crt
```

# Configure DNS

- DNS is used to find XMPP servers.
- A record is enough for a simple setup
- SRV records for a more complex setup
  - Another host or another port
  - Don't point to a CNAME RR
  - Priority and fallback

## Example A record

```
jabber.motd.ch.      A      5.45.105.71
```

## Example SRV record

```
_xmpp-client._tcp.jabber.motd.com. 18000 IN SRV 0 5 5222 xmpp.motd.com.  
_xmpp-server._tcp.jabber.motd.com. 18000 IN SRV 0 5 5269 xmpp.motd.com.
```

## Activate Configuration

```
# ln -s /etc/prosody/conf.{avail,d}/jabber.motd.ch.cfg.lua
```

## Start Prosody:

```
# service prosody restart
```

## Check logfiles:

```
# tail -n0 -f /var/log/prosody/prosody.{log,err}
[...]
[...] jabber.motd.ch:posix info Successfully daemonized to PID 11588
[...]
```



# Configure Firewall

Which ports is Prosody listening to?

```
# lsof -u prosody -sTCP:LISTEN | grep TCP
lua5.1  prosody  IPv6 [...] TCP *:xmpp-client (LISTEN)
lua5.1  prosody  IPv4 [...] TCP *:xmpp-client (LISTEN)
lua5.1  prosody  IPv6 [...] TCP *:xmpp-server (LISTEN)
lua5.1  prosody  IPv4 [...] TCP *:xmpp-server (LISTEN)
```

Which are following ports:

```
$ grep -E xmpp-.*tcp /etc/services
xmpp-client 5222/tcp jabber-client # Jabber Client Connection
xmpp-server 5269/tcp jabber-server # Jabber Server Connection
```

Firewall configuration for IPv4 and IPv6:

- 5222/tcp for incoming client to server connections
- 5269/tcp for incoming and outgoing server to server connections

# User Accounts

- Unique XMPP Address (Jabber ID, JID): `emanuel.duss@jabber.motd.ch`
- Multiple logins with priority: `emanuel.duss@jabber.motd.ch/mobile`
- Set authentication = "internal\_hashed" in `prosody.cfg.lua`
- Contact list in `/var/lib/prosody/jabber%2emotd%2ech/roster/`

Add a new user:

```
# prosodyctl adduser emanuel.duss@jabber.motd.ch
```

Password hashed in `/var/lib/prosody/jabber%2emotd%2ech/`

```
return {  
    ["iteration_count"] = 4096;  
    ["stored_key"] = "e9830c4a367d9e94ece9d5da96e526215131981e";  
    ["salt"] = "a3cbe835-fb16-487e-822c-7946b97b401b";  
    ["server_key"] = "be197281c52acdd3b1aa5eecbe5718f9545bac20";  
};
```

# Check your server configuration

- Use <http://xmpp.net> to check your XMPP and TLS configuration.

## Score

xmpp.motd.ch:5222



Grade:

A

Modify Account

Basic | Advanced | Proxy | Voice and Video

**Login Options**

Protocol:

Username:

Domain:

Resource:

Password:


Remember password

**User Options**

Local alias:

New mail notifications

Use this buddy icon for this account:



Create this new account on the server

- WP “XMPP”: <https://en.wikipedia.org/wiki/XMPP>
- OTR: <https://otr.cyberpunks.ca/>
- Prosody Documentation: <https://prosody.im/doc/ce> (Really good!)

?