

Table of Contents

Introduction	1
Note	1
Prerequisites	1
Installing ZeroTier	1
On your Synapse server	1
On your Mac	2
Set up the network	2
Install Barcelona	2
Install the bridge	2
Register the App service on your Synapse server	3
Installing wsproxy	3
Test	4
Start wsproxy and the bridge automatically	4
Test again	6

Last update:
2022/12/12 11:45

advanced:imessage_bridge https://familybrown.org/dokuwiki/doku.php?id=advanced:imessage_bridge&rev=1670845503

Introduction

These notes discuss the installation of the [Mautrix iMessage bridge](#). This will involve installing a websocket proxy and the bridge component on a Mac (or other system running macOS), and setting up a VPN between the Mac and your Synapse server using ZeroTier.

Note

This bridge, at least as configured following these instructions, can only bridge a single iMessage user. You may be able to adapt these instructions to bridge to multiple users, but I can't give any assistance on that area, at least at this time.

Prerequisites

- You've installed Synapse on a public server using the Ansible playbook at <https://github.com/spantaleev/matrix-docker-ansible-deploy>, and you have root access to that server
- You have an available system running macOS Big Sur (11.x) or newer
 - Because the bridge will run on this system, it will need to be running 24x7x365
 - You'll need to disable SIP and AMFI as described at <https://docs.mau.fi/bridges/go/imessage/mac-nosip/setup.html#disabling-sip-and-amfi>
 - Since this disables significant security features, it's best this macOS installation be on its own machine or VM. macOS runs well under Proxmox; see <https://github.com/luchina-gabriel/OSX-PROXMOX>
 - You also need to be logged into iMessage on that system.
 - The macOS system must have Xcode installed (free from the App Store) along with [Homebrew](#)

Installing ZeroTier

On your Synapse server

```
curl -s
'https://raw.githubusercontent.com/zerotier/ZeroTierOne/master/doc/contact%40ze
rotier.com.gpg' | gpg --import && \ if z=$(curl -s
'https://install.zerotier.com/' | gpg); then echo "$z" | sudo bash; fi
```

On your Mac

Download the installer from zerotier.com and install it

Set up the network

- Log in to your account on zerotier.com—create one if you don't have it; it's free.
- Go to the Networks tab, and click "Create A Network"
- Once the network is created, you can optionally (but recommended) change the name to something meaningful. Make a note of the network ID.
- On the Mac, go to the ZeroTier menu, select "Join New Network...", and paste in the network ID.
- On your Synapse server, run `sudo zerotier-cli join <network-id>`
- Go back to the Networks tab on zerotier.com, and click on the network you just created. You'll see two Members; check the box under "Auth?" to allow them to join. In a moment, you'll see managed IPs show up for each member. Give each member a name so you can readily tell them apart, and test connectivity by pinging each one from the other.

Install Barcelona

Barcelona is a component used by the bridge to interface with the iMessage service, and it will need to be built from source code. You'll need to run these commands from your preferred terminal on the Mac:

- `mkdir -p ~/src/{mautrix-imessage,mautrix-wsproxy,barcelona-mautrix}`
- `brew install xcodegen && sudo gem install xcpretty`
- `cd ~`
- `git clone https://github.com/beeper/barcelona`
- `cd barcelona`
- `make mautrix-macos`
- `cp ~/barcelona/Build/macOS/Build/Products/Release/barcelona-mautrix-macOS ~/src/barcelona-mautrix/`
- `sudo cp ~/barcelona/com.apple.security.xpc.plist /Library/Preferences/`

Install the bridge

This step is done on the Mac. First, browse to <https://mau.dev/mautrix/imessage/pipelines?scope=branches&page=1>. Download the latest (i.e., first on the list) build from the master branch, with the appropriate architecture. If you don't know, download the one that says `build universal:archive`. Unzip it.

- Move the contents to `~/src/mautrix-imessage/`, and rename `example-config.yaml` to

config.yaml.

- Edit config.yaml.
 - In the homeserver: section, change address: to point to your Synapse server. Change websocket_proxy to ws://localhost:29331. Change domain: to the domain of your Synapse server.
 - In the imessage: section, change platform: to mac-nosip and imessage_rest_path: to /Users/<you>/src/barcelona-mautrix/barcelona-mautrix, where <you> is your username on the Mac.
 - In the bridge: section, change user: to your user. Change login_shared_secret: to the value of matrix_synapse_ext_password_provider_shared_secret_auth_shared_secret: in your Ansible vars.yaml file.
 - In bridge -> encryption, change allow: to true, and default: to true
 - Save and close the file.
- Go to the terminal of your preference, run `cd ~/src/mautrix-imessage/`, followed by `./mautrix-imessage -g`. This will generate registration.yaml in that directory. Edit that file.
- In registration.yaml, change url: to `http://<zerotier_ip>:29331`, where <zerotier_ip> is the IP address of your Mac in the ZeroTier network.

Register the App service on your Synapse server

- SSH in to your Synapse server as root
- `mkdir -p /matrix/mautrix-imessage/config`
- `nano /matrix/mautrix-imessage/config/registration.yaml`
- Paste in the edited contents of registration.yaml from your Mac.

Installing wsproxy

Compiled binaries for macOS aren't available for download for wsproxy, so you'll need to compile it yourself. To do this, first go to <https://go.dev> and download (and install) Go for macOS. Then:

- `cd ~`
- `git clone https://github.com/mautrix/wsproxy`
- `cd wsproxy`
- `go build -o mautrix-wsproxy`
- `mv mautrix-wsproxy ~/src/mautrix-wsproxy`
- `cp example-config.yaml ~/src/mautrix-wsproxy/config.yaml`
- Edit config.yaml
 - Set listen_address: to 0.0.0.0:29331
 - Replace the value of as with the value of as_token from registration.yaml
 - Replace the value of hs with the value of hs_token from registration.yaml
 - Remove everything after the hs: line

Configure Synapse to register this bridge These steps need to be taken on whatever machine you're using to run the Ansible playbook. First, edit vars .yaml, and add the following to the end:

```
# App service registration file for maatrix-imessage
matrix_synapse_container_extra_arguments:
- '-mount type=bind,src=/matrix/maatrix-
imessage/config/registration.yaml,dst=/matrix-maatrix-imessage-
registration.yaml,ro'
matrix_synapse_app_service_config_files:
- /matrix-maatrix-imessage-registration.yaml
```

Then, re-run the playbook with `ansible-playbook -i inventory/hosts setup.yml --tags=setup-all,start`.

Test

Once that completes, test it out quickly. Back on the Mac, open two terminal windows. In the first, run `~/src/maatrix-wsproxy/maatrix-wsproxy`. In the second, run `~/src/maatrix-imessage/maatrix-imessage`. If the latter starts up successfully, press Ctrl-C to terminate each of them.

Start wsproxy and the bridge automatically

Back on the Mac, create `~/maatrix-imessage.plist`. Its contents should be:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">

<plist version="1.0">
<dict>

  <key>Label</key>
  <string>com.github.maatrix-imessage</string>

  <key>WorkingDirectory</key>
  <string>/Users/<you>/src/maatrix-imessage</string>

  <key>ProgramArguments</key>
  <array>
  <string>./maatrix-imessage</string>
  <string>--config</string>
```

```

    <string>config.yaml</string>
  </array>

  <key>RunAtLoad</key>
  <true/>

  <key>KeepAlive</key>
  <true/>

</dict>
</plist>

```

...where <you> is your username on the Mac. Then create ~/mautrix-wsproxy.plist. Its contents should be:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">

<plist version="1.0">
<dict>

  <key>Label</key>
  <string>com.github.mautrix-wsproxy</string>

  <key>WorkingDirectory</key>
  <string>/Users/<you>/src/mautrix-wsproxy</string>

  <key>ProgramArguments</key>
  <array>
  <string>./mautrix-wsproxy</string>
  <string>-config</string>
  <string>config.yaml</string>
  </array>

  <key>RunAtLoad</key>
  <true/>

  <key>KeepAlive</key>
  <true/>

</dict>
</plist>

```

Then, at your favorite terminal: * mkdir ~/Library/LaunchAgents * cp ~/.plist ~/Library/LaunchAgents/ * launchctl load mautrix-wsproxy.plist * launchctl load mautrix-imessage.plist

Test again

Log in to your homserver using whatever client app you like, and send a message to @imessagebot:example.com, where example.com is the domain of your homserver.

From:

<https://familybrown.org/dokuwiki/> - danb35's Wiki

Permanent link:

https://familybrown.org/dokuwiki/doku.php?id=advanced:imessage_bridge&rev=1670845503

Last update: **2022/12/12 11:45**

