title: Hosting your own SMP Server

revision: 31.07.2023

| Updated 05.06.2023 | Languages: EN, FR, CZ |

Hosting your own SMP Server

Overview

SMP server is the relay server used to pass messages in SimpleX network. SimpleX Chat apps have preset servers (for mobile apps these are smp11, smp12 and smp14.simplex.im), but you can easily change app configuration to use other servers.

SimpleX clients only determine which server is used to receive the messages, separately for each contact (or group connection with a group member), and these servers are only temporary, as the delivery address can change.

Please note: when you change the servers in the app configuration, it only affects which server will be used for the new contacts, the existing contacts will not automatically move to the new servers, but you can move them manually using "Change receiving address" button in contact/member information pages – it will be automated soon.

Installation

1. First, install `smp-server`:

   - Manual deployment (see below)

   - Semi-automatic deployment:

     - Offical installation script
     - Docker container
     - Linode StackScript

Manual installation requires some preliminary actions:

1. Install binary:

   - Using offical binaries:

     ```
     sh curl -L https://github.com/simplex-chat/simplexmq/
     releases/latest/download/smp-server-ubuntu-20_04-x86-64
     -o /usr/local/bin/smp-server
     ```

   - Compiling from source:

Please refer to Build from source: Using your distribution

2. Create user and group for `smp-server`:

   ```
   sh sudo useradd -m smp
   ```

3. Create necessary directories and assign permissions:

   ```
   sh sudo mkdir -p /var/opt/simplex /etc/opt/simplex sudo chown
   smp:smp /var/opt/simplex /etc/opt/simplex
   ```

4. Allow `smp-server` port in firewall:

   ~~~sh

   # For Ubuntu

   sudo ufw allow 5223/tcp

   # For Fedora

   sudo firewall-cmd --permanent --add-port=5223/tcp && \ sudo firewall-cmd --reload ```

5. **Optional** — If you're using distribution with `systemd`, create `/etc/systemd/system/smp-server.service` file with the following content:

   ```sh [Unit] Description=SMP server systemd service

   [Service] User=smp Group=smp Type=simple ExecStart=/usr/local/bin/smp-server start +RTS -N -RTS ExecStopPost=/usr/bin/env sh -c '[ -e "/var/opt/simplex/smp-server-store.log" ] && cp "/var/opt/simplex/smp-server-store.log" "/var/opt/simplex/smp-server-store.log.bak"' LimitNOFILE=65535 KillSignal=SIGINT TimeoutStopSec=infinity

   [Install] WantedBy=multi-user.target ```

   And execute `sudo systemctl daemon-reload`.

## Tor installation

smp-server can also be deployed to serve from tor network. Run the following commands as `root` user.

1. Install tor:

   We're assuming you're using Ubuntu/Debian based distributions. If not, please refer to offical tor documentation or your distribution guide.

   - Configure offical Tor PPA repository:
   ~~~

```
sh CODENAME="$(lsb_release -c | awk '{print $2}')" echo "deb [signed-by=/usr/share/keyrings/tor-archive-keyring.gpg] https://deb.torproject.org/torproject.org ${CODENAME} main deb-src [signed-by=/usr/share/keyrings/tor-archive-keyring.gpg] https://deb.torproject.org/torproject.org ${CODENAME} main" > /etc/apt/sources.list.d/tor.list
```

- Import repository key:

```
sh curl --proto '=https' --tlsv1.2 -sSf https://deb.torproject.org/torproject.org/A3C4F0F979CAA22CDBA8F512EE8CBC9E886DDD89.asc | gpg --dearmor | tee /usr/share/keyrings/tor-archive-keyring.gpg >/dev/null
```

- Update repository index:

```
sh apt update
```

- Install `tor` package:

```
sh apt install -y tor deb.torproject.org-keyring
```

2. Configure tor:

- File configuration:

Open tor configuration with your editor of choice (`nano`,`vim`,`emacs`,etc.):

```
sh vim /etc/tor/torrc
```

And insert the following lines to the bottom of configuration. Please note lines starting with #: this is comments about each individual options.

\`\`\`sh

Enable log (otherwise, tor doesn't seemd to deploy onion address)

Log notice file /var/log/tor/notices.log

Enable single hop routing (2 options below are dependencies of third). Will reduce latency in exchange of anonimity (since tor runs alongside smp-server and onion address will be displayed in clients, this is totally fine)

SOCKSPort 0 HiddenServiceNonAnonymousMode 1 HiddenServiceSingleHopMode 1

smp-server hidden service host directory and port mappings

HiddenServiceDir /var/lib/tor/simplex-smp/ HiddenServicePort 5223 localhost:5223 ```

- Create directories:

  ```
  sh mkdir /var/lib/tor/simplex-smp/ && chown debian-tor:debian-tor /var/lib/tor/simplex-smp/ && chmod 700 /var/lib/tor/simplex-smp/
  ```

3. Start tor:

   Enable `systemd` service and start tor. Offical `tor` is a bit flunky on the first start and may not create onion host address, so we're restarting it just in case.

   ```
   sh systemctl enable tor && systemctl start tor && systemctl restart tor
   ```

4. Display onion host:

   Execute the following command to display your onion host address:

   ```
   sh cat /var/lib/tor/simplex-smp/hostname
   ```

Configuration

To see which options are available, execute `smp-server` without flags:

```sh sudo su smp -c smp-server

... Available commands: init Initialize server - creates /etc/opt/simplex and /var/opt/simplex directories and configuration files start Start server (configuration: /etc/opt/simplex/smp-server.ini) delete Delete configuration and log files ```

You can get further help by executing `sudo su smp -c "smp-server <command> -h"`

After that, we need to configure `smp-server`:

## Interactively

Execute the following command:

```
sh sudo su smp -c "smp-server init"
```

There are several options to consider:

- `Enable store log to restore queues and messages on server restart (Yn):`

  Enter `y` to enable saving and restoring connections and messages when the server is restarted.

  Please note: it is important to use SIGINT to restart the server, as otherwise the undelivered messages will not be restored. The connections will be restored irrespective of how the server is restarted, as unlike messages they are added to append-only log on every change.

- `Enable logging daily statistics (yN):`

  Enter `y` to enable logging statistics in CSV format, e.g. they can be used to show aggregate usage charts in `Grafana`.

These statistics include daily counts of created, secured and deleted queues, sent and received messages, and also daily, weekly, and monthly counts of active queues (that is, the queues that were used for any messages). We believe that this information does not include anything that would allow correlating different queues as belonging to the same users, but please let us know, confidentially, if you believe that this can be exploited in any way.

- `Require a password to create new messaging queues?`

  Enter `r` or your arbitrary password to password-protect `smp-server`, or `n` to disable password protection.

- `Enter server FQDN or IP address for certificate (127.0.0.1):`
```

> Enter your domain or ip address that your smp-server is running on - it will be included in server certificates and also printed as part of server address.

### Via command line options

Execute the following command:

```sh sudo su smp -c "smp-server init -h"

... Available options: -l,--store-log Enable store log for persistence -s,--daily-stats Enable logging daily server statistics -a,--sign-algorithm ALG Signature algorithm used for TLS certificates: ED25519, ED448 (default: ED448) --ip IP Server IP address, used as Common Name for TLS online certificate if FQDN is not supplied (default: "127.0.0.1") -n,--fqdn FQDN Server FQDN used as Common Name for TLS online certificate --no-password Allow creating new queues without password --password PASSWORD Set password to create new messaging queues -y,--yes Non-interactive initialization using command-line options -h,--help Show this help text ```

You should determine which flags are needed for your use-case and then execute `smp-server init` with `-y` flag for non-interactive initialization:

```
sh sudo su smp -c "smp-server init -y -<your flag> <your option>"
```

For example, run:

```
sh sudo su smp -c "smp-server init -y -l --ip 192.168.1.5 --password test"
~~~

to initialize your `smp-server` configuration with:

- restoring connections and messages when the server is restarted (`-l` flag),
- IP address `192.168.1.5`,
- protect `smp-server` with a password `test`.

---

After that, your installation is complete and you should see in your teminal output something like this:

```sh Certificate request self-signature ok subject=CN = 127.0.0.1 Server initialized, you can modify configuration in /etc/opt/simplex/smp-server.ini.

# Run `smp-server start` to start server.

You should store CA private key securely and delete it from the server. If server TLS credential is compromised this key can be used to sign a new one, keeping the same server identity and established connections.
~~~

# CA private key location: /etc/opt/simplex/ca.key

SMP server v3.4.0 Fingerprint: d5fcsc7hhtPpexYUbI2XPxDbyU2d3WsVmROimcL90ss= Server address: smp://d5fcsc7hhtPpexYUbI2XPxDbyU2d3WsVmROimcL90ss=:V8ONoJ6ICwnrZnTCQuSHfCEY
ve8=@ ```

The server address above should be used in your client configuration and if you added server password it should only be shared with the other people when you want to allow them to use your server to receive the messages (all your contacts will be able to send messages, as it does not require a password). If you passed IP address or hostnames during the initialisation, they will be printed as part of server address, otherwise replace `<hostnames>` with the actual server addresses.

# Documentation

All necessary files for `smp-server` are located in `/etc/opt/simplex/` folder.

Stored messages, connections, statistics and server log are located in `/var/opt/simplex/` folder.

## SMP server address

SMP server address has the following format:

```
smp://<fingerprint>[:<password>]@<public_hostname>[,<onion_hostname>]
```

- `<fingerprint>`

  Your `smp-server` fingerprint of certificate. You can check your certificate fingerprint in `/etc/opt/simplex/fingerprint`.

- **optional** `<password>`

  Your configured password of `smp-server`. You can check your configured pasword in `/etc/opt/simplex/smp-server.ini`, under `[AUTH]` section in `create_password:` field.

- `<public_hostname>`, **optional** `<onion_hostname>`

  Your configured hostname(s) of `smp-server`. You can check your configured hosts in `/etc/opt/simplex/smp-server.ini`, under `[TRANSPORT]` section in `host:` field.

Systemd commands

To start `smp-server` on host boot, run:

```sh
sudo systemctl enable smp-server.service

Created symlink /etc/systemd/system/multi-user.target.wants/smp-
server.service → /etc/systemd/system/smp-server.service.
```

To start `smp-server`, run:

```
sh sudo systemctl start smp-server.service
```

To check status of `smp-server`, run:

```sh
sudo systemctl status smp-server.service

● smp-server.service - SMP server Loaded: loaded (/etc/systemd/system/
smp-server.service; enabled; vendor preset: enabled) Active: active (running)
since Sat 2022-11-23 19:23:21 UTC; 1min 48s ago Main PID: 30878 (smp-
server) CGroup: /docker/
5588ab759e80546b4296a7c50ffebbb1fb7b55b8401300e9201313b720989aa8/
system.slice/smp-server.service └─30878 smp-server start

Nov 23 19:23:21 5588ab759e80 systemd[1]: Started SMP server. Nov 23
19:23:21 5588ab759e80 smp-server[30878]: SMP server v3.4.0 Nov 23
19:23:21 5588ab759e80 smp-server[30878]: Fingerprint:
d5fcsc7hhtPpexYUbI2XPxDbyU2d3WsVmROimcL90ss= Nov 23 19:23:21
5588ab759e80 smp-server[30878]: Server address: smp://
d5fcsc7hhtPpexYUbI2XPxDbyU2d3WsVmROimcL90ss=:V8ONoJ6ICwnrZnTCQuSHfCEY
ve8=@ Nov 23 19:23:21 5588ab759e80 smp-server[30878]: Store log: /var/
opt/simplex/smp-server-store.log Nov 23 19:23:21 5588ab759e80 smp-
server[30878]: Listening on port 5223 (TLS)... Nov 23 19:23:21
5588ab759e80 smp-server[30878]: not expiring inactive clients Nov 23
19:23:21 5588ab759e80 smp-server[30878]: creating new queues requires
password
```

To stop `smp-server`, run:

```
sh sudo systemctl stop smp-server.service
```

To check tail of `smp-server` log, run:

```sh
sudo journalctl -fu smp-server.service

Nov 23 19:23:21 5588ab759e80 systemd[1]: Started SMP server. Nov 23
19:23:21 5588ab759e80 smp-server[30878]: SMP server v3.4.0 Nov 23
19:23:21 5588ab759e80 smp-server[30878]: Fingerprint:
d5fcsc7hhtPpexYUbI2XPxDbyU2d3WsVmROimcL90ss= Nov 23 19:23:21
5588ab759e80 smp-server[30878]: Server address: smp://
d5fcsc7hhtPpexYUbI2XPxDbyU2d3WsVmROimcL90ss=:V8ONoJ6ICwnrZnTCQuSHfCEY
ve8=@ Nov 23 19:23:21 5588ab759e80 smp-server[30878]: Store log: /var/
opt/simplex/smp-server-store.log Nov 23 19:23:21 5588ab759e80 smp-
server[30878]: Listening on port 5223 (TLS)... Nov 23 19:23:21
5588ab759e80 smp-server[30878]: not expiring inactive clients Nov 23
19:23:21 5588ab759e80 smp-server[30878]: creating new queues requires
password
```

Monitoring

You can enable `smp-server` statistics for `Grafana` dashboard by setting value `on` in `/etc/opt/simplex/smp-server.ini`, under `[STORE_LOG]` section in `log_stats:` field.

Logs will be stored as `csv` file in `/var/opt/simplex/smp-server-stats.daily.log`. Fields for the `csv` file are:

```
sh
fromTime,qCreated,qSecured,qDeleted,msgSent,msgRecv,dayMsgQueues,weekMsgQu
```

- `fromTime` - timestamp; date and time of event
- `qCreated` - int; created queues
- `qSecured` - int; established queues
- `qDeleted` - int; deleted queues
- `msgSent` - int; sent messages
- `msgRecv` - int; received messages
- `dayMsgQueues` - int; active queues in a day
- `weekMsgQueues` - int; active queues in a week
- `monthMsgQueues` - int; active queues in a month

To import `csv` to `Grafana` one should:

1. Install Grafana plugin: [Grafana - CSV datasource]

2. Allow local mode by appending following:

    ```
    sh [plugin.marcusolsson-csv-datasource] allow_local_mode =
    true
    ```

    ... to `/etc/grafana/grafana.ini`

3. Add a CSV data source:

    - In the side menu, click the Configuration tab (cog icon)
    - Click Add data source in the top-right corner of the Data Sources tab
    - Enter "CSV" in the search box to find the CSV data source
    - Click the search result that says "CSV"
    - In URL, enter a file that points to CSV content

4. You're done! You should be able to create your own dashboard with statistics.

For further documentation, see: CSV Data Source for Grafana - Documentation

Configuring the app to use the server

To configure the app to use your messaging server copy it's full address, including password, and add it to the app. You have an option to use your server together with preset servers or without them - you can remove or disable them.

It is also possible to share the address of your server with your friends by letting them scan QR code from server settings - it will include server password, so they will be able to receive messages via your server as well.

Please note: you need SMP server version 4.0 to have password support. If you already have a deployed server, you can add password by adding it to server INI file.