

---

# **synadm**

***Release 0.35***

**Johannes Tiefenbacher**

**Jul 01, 2022**



**CONTENTS:**

<b>1</b>	<b>Command Line Reference</b>	<b>1</b>
1.1	Main Command . . . . .	1
1.2	Config . . . . .	2
1.3	User . . . . .	3
1.4	Room . . . . .	10
1.5	Media . . . . .	16
1.6	History . . . . .	19
1.7	Group . . . . .	20
1.8	Matrix . . . . .	20
1.9	Regtok . . . . .	22
<b>2</b>	<b>Python Package Documentation</b>	<b>25</b>
2.1	Synadm CLI Package . . . . .	25
2.2	Synadm Package . . . . .	26
<b>3</b>	<b>Indices and tables</b>	<b>37</b>
	<b>Python Module Index</b>	<b>39</b>
	<b>Index</b>	<b>41</b>



## COMMAND LINE REFERENCE

### 1.1 Main Command

#### 1.1.1 synadm

the Matrix-Synapse admin CLI

```
synadm [OPTIONS] COMMAND [ARGS]...
```

#### Options

**-v, --verbose**

Enable INFO (-v) or DEBUG (-vv) logging on console.

**--batch, --no-batch**

Enable batch behavior (no interactive prompts).

**-o, --output <output>**

Override default output format.

#### Options

yaml | json | human | pprint | y | j | h | p |

**-c, --config-file <config\_file>**

Configuration file path.

#### Default

~/.config/synadm.yaml

#### Commands

**config**

Modify synadm's configuration.

**group**

Manage groups (communities)

**history**

Purge historic events from Synapse database

**matrix**

Execute Matrix API calls.

**media**

Manage local and remote media

**regtok**

Manage registration tokens

**room**

Manipulate rooms and room membership

**user**

List, add, modify, deactivate/erase users,...

**version**

Get the Synapse server version.

## 1.2 Config

### 1.2.1 synadm config

Modify synadm's configuration. Configuration details are generally always asked interactively. Command line options override the suggested defaults in the prompts.

```
synadm config [OPTIONS]
```

#### Options

**-u, --user** <user>

Admin user allowed to access the Synapse admin API's.

**-t, --token** <token>

The Admin user's access token.

**-b, --base-url** <base\_url>

The base URL Synapse is running on. Typically this is <https://localhost:8008> or <https://localhost:8448>. If Synapse is configured to expose its admin API's to the outside world it might as well be something like this: <https://example.org:8448>

**-p, --admin-path** <admin\_path>

The path Synapse provides its admin API's, usually the default fits most installations.

**-m, --matrix-path** <matrix\_path>

The path Synapse provides the regular Matrix API's, usually the default fits most installations.

**-w, --timeout** <timeout>

The time in seconds synadm should wait for responses from admin API's or Matrix API's. The default is 7 seconds.

**-o, --output <output>**

How synadm displays data by default. ‘human’ gives a tabular or list view depending on the fetched data. This mode needs your terminal to be quite wide! ‘json’ displays exactly as the API responded. ‘pprint’ shows nicely formatted json. ‘yaml’ is the currently recommended output format. It doesn’t need as much terminal width as ‘human’ does. Note that the default output format can always be overridden by using global switch -o (eg ‘synadm -o pprint user list’).

**Options**

yaml | json | human | pprint

**-d, --server-discovery <server\_discovery>**

FIXME The method used for server discovery. This can either be ‘local’ or ‘dns’. The ‘local’ mode is using the ‘well-known’ file of your server, if present. The ‘dns’ mode is using the SRV record of your domain. FIXME

**Options**

well-known | dns

**-n, --homeserver <homeserver>**

Synapse homeserver hostname. Usually matrix.DOMAIN or DOMAIN. The default value ‘auto-retrieval’ will try to discover the name using the method selected by `--server-discovery`.

## 1.3 User

### 1.3.1 synadm user

List, add, modify, deactivate/erase users, reset passwords

```
synadm user [OPTIONS] COMMAND [ARGS]...
```

#### deactivate

Deactivate or gdpr-erase a user. Provide matrix user ID (@user:server) as argument. It removes active access tokens, resets the password, and deletes third-party IDs (to prevent the user requesting a password reset).

```
synadm user deactivate [OPTIONS] USER_ID
```

#### Options

**-e, --gdpr-erase**

Marks the user as GDPR-erased. This means messages sent by the user will still be visible by anyone that was in the room when these messages were sent, but hidden from users joining the room afterwards.

**Default**

False

## Arguments

### USER\_ID

Required argument

## details

View details of a user account.

```
synadm user details [OPTIONS] USER_ID
```

## Arguments

### USER\_ID

Required argument

## list

List and search for users

```
synadm user list [OPTIONS]
```

## Options

**-f, --from <from\_>**

Offset user listing by given number. This option is also used for pagination.

**Default**

0

**-l, --limit <limit>**

Limit user listing to given number

**Default**

100

**-g, --guests, -G, --no-guests**

Show guest users.

**-d, --deactivated**

Also show deactivated/erased users

**Default**

False

**-n, --name <name>**

Search users by name - filters to only return users with user ID localparts or displaynames that contain this value (localpart is the left part before the colon of the matrix ID (@user:server))

**-i, --user-id <user\_id>**

Search users by ID - filters to only return users with Matrix IDs (@user:server) that contain this value



## login

Get an access token for a given user.

Useful for when admins wish to do actions on behalf of a user.

If no `--expire*` option is given, a default token expiry time of exactly 1 day (24h) is used. If it's desired that the token never expires, use `--expire-never`

This API does not generate a new device for the user, and so will not appear in their `/devices` list, and in general the target user should not be able to tell they have been logged in as.

To expire the token before the expiry date/time is reached, call the standard `/logout` API with the token. Note: The token will expire if the admin user calls `/logout/all` from any of their devices, but the token will not expire if the target user does the same.

```
synadm user login [OPTIONS] USER_ID
```

## Options

**-d, --expire-days <expire\_days>**

Expire token after this number of days.

**--expire <expire>**

Expire token after this point in time. Eg. '2021-01-01', see above for available date/time formats.

**--expire-ts <expire\_ts>**

Expire token after this point in time giving a unix timestamp in ms.

**--expire-never**

Never expire token.

**Default**

False

## Arguments

**USER\_ID**

Required argument

## media

List all local media uploaded by a user.

Provide matrix user ID (`@user:server`) as argument.

Gets a list of all local media that a specific `user_id` has created. By default, the response is ordered by descending creation date and ascending media ID. The newest media is on top. You can change the order with options `--order-by` and `--reverse`.

Caution. The database only has indexes on the columns `media_id`, `user_id` and `created_ts`. This means that if a different sort order is used (`upload_name`, `last_access_ts`, `media_length`, `media_type`, `quarantined_by` or `safe_from_quarantine`), this can cause a large load on the database, especially for large environments

```
synadm user media [OPTIONS] USER_ID
```

## Options

**-f, --from** <from>

Offset media listing by given number. This option is also used for pagination.

**Default**

0

**-l, --limit** <limit>

Limit media listing to given number

**Default**

100

**-s, --sort** <sort>

The method by which to sort the returned list of media. If the ordered field has duplicates, the second order is always by ascending media\_id, which guarantees a stable ordering.

**Options**

media\_id | upload\_name | created\_ts | last\_access\_ts | media\_length | media\_type | quarantined\_by | safe\_from\_quarantine

**-r, --reverse**

Direction of media order. If set it will reverse the sort order of --order-by method.

**--datetime, --dt, --timestamp, --ts**

Display created and last accessed timestamps in a human readable format, or as a unix timestamp in milliseconds. [default: datetime].

## Arguments

**USER\_ID**

Required argument

## membership

List all rooms a user is member of.

Provide matrix user ID (@user:server) as argument.

`synadm user membership [OPTIONS] USER_ID`

## Options

**--aliases, --ids**

Display rooms as canonical aliases or room ID's. [default: aliases]

## Arguments

### USER\_ID

Required argument

## modify

Create or modify a local user. Provide matrix user ID (@user:server) as argument.

```
synadm user modify [OPTIONS] USER_ID
```

## Options

### -p, --password-prompt

Set password interactively.

### -P, --password <password>

Set password on command line.

### -n, --display-name <display\_name>

Set display name. defaults to the value of user\_id

### -t, --threepid <threepid>

Add a third-party identifier. This can be an email address or a phone number. Threepids are used for several things: For use when logging in, as an alternative to the user id. In the case of email, as an alternative contact to help with account recovery, as well as to receive notifications of missed messages. Format: medium value (eg. `--threepid email <user@example.org>`). This option can also be stated multiple times, i.e. a user can have multiple threepids configured.

### -v, --avatar-url <avatar\_url>

Set avatar URL. Must be a MXC URI ([https://matrix.org/docs/spec/client\\_server/r0.6.0#matrix-content-mxc-uris](https://matrix.org/docs/spec/client_server/r0.6.0#matrix-content-mxc-uris))

### -a, --admin, -u, --no-admin

Grant user admin permission. Eg user is allowed to use the admin API.

### --activate

Re-activate user.

### --deactivate

Deactivate user. Use with caution! Deactivating a user removes their active access tokens, resets their password, kicks them out of all rooms and deletes third-party identifiers (to prevent the user requesting a password reset). See also “user deactivate” command.

## Arguments

### USER\_ID

Required argument

## password

Change a user's password.

To prevent the user from being logged out of all sessions use option -n.

```
synadm user password [OPTIONS] USER_ID
```

## Options

### -n, --no-logout

Don't log user out of all sessions on all devices.

### -p, --password <password>

New password.

## Arguments

### USER\_ID

Required argument

## prune-devices

Delete devices and invalidate access tokens of a user.

Deletes devices of a user and invalidates any access token associated with them. Starts from deleting the oldest devices, not seen in a number of days, which may be abandoned.

Note that this will affect the encryption and decryption of messages sent by other users to this user or to rooms where the user is present.

```
synadm user prune-devices [OPTIONS] USER_ID
```

## Options

### -l, --list-only

Dry-run: does not perform the deletion but shows what would be done. If you want to list all devices/sessions, you can also use the whois command.

#### Default

False

### -d, --min-days <min\_days>

How many days need to have passed from the last time a device was seen for it to be deleted.

#### Default

90

**-s, --min-surviving** <min\_surviving>

Stop processing devices when only this number of devices is left for this user. Allows to reduce disruption by preserving recently used devices/sessions.

**Default**

1

**-i, --device-id** <device\_id>

Only search devices with this ID. The other options still apply if they're not 0.

**--datetime, --dt, --timestamp, --ts**

Display 'last seen date/time' in a human readable format, or as a unix timestamp in milliseconds. [default: datetime].

## Arguments

**USER\_ID**

Required argument

## search

A shortcut to 'synadm user list -d -g -n <search-term>'.

Searches for users by name/matrix-ID, including deactivated users as well as guest users. Also, compared to the original command, a case-insensitive search is done.

```
synadm user search [OPTIONS] SEARCH_TERM
```

## Options

**-f, --from** <from\_>

Offset user listing by given number. This option is also used for pagination.

**Default**

0

**-l, --limit** <limit>

Maximum amount of users to return.

**Default**

100

## Arguments

**SEARCH\_TERM**

Required argument

## shadow-ban

Shadow-ban or unban a user.

Useful for moderating malicious or egregiously abusive users.

A shadow-banned user will not receive any notification, but still will be unable to contact anyone on the server. Use this as a tool of last resort since it may lead to confusing or broken behaviour for the client.

Generally, it is more appropriate to ban or kick abusive users.

```
synadm user shadow-ban [OPTIONS] USER_ID
```

## Options

**-u, --unban**

Unban the specified user

## Arguments

**USER\_ID**

Required argument

## whois

Return information about the active sessions for a specific user

```
synadm user whois [OPTIONS] USER_ID
```

## Arguments

**USER\_ID**

Required argument

# 1.4 Room

## 1.4.1 synadm room

Manipulate rooms and room membership

The syntax of room IDs in synadm is as the Matrix spec defines: <https://spec.matrix.org/latest/#room-structure>. Make sure to escape the ! character from your shell. In bash and zsh this can be achieved by using single quotes (‘), eg. ‘!id123abc:matrix.DOMAIN’

```
synadm room [OPTIONS] COMMAND [ARGS]...
```

## delete

Delete and possibly purge a room

```
synadm room delete [OPTIONS] ROOM_ID
```

## Options

**-u, --new-room-user-id** <new\_room\_user\_id>

If set, a new room will be created with this user ID as the creator and admin, and all users in the old room will be moved into that room. If not set, no new room will be created and the users will just be removed from the old room. The user ID must be on the local server, but does not necessarily have to belong to a registered user.

**-n, --room-name** <room\_name>

A string representing the name of the room that new users will be invited to. Defaults to “Content Violation Notification”

**-m, --message** <message>

A string containing the first message that will be sent as new\_room\_user\_id in the new room. Ideally this will clearly convey why the original room was shut down. Defaults to “Sharing illegal content on this server is not permitted and rooms in violation will be blocked.”

**-b, --block**

If set, this room will be added to a blocking list, preventing future attempts to join the room

**Default**

False

**--no-purge**

Prevent removing of all traces of the room from your database.

**Default**

False

## Arguments

**ROOM\_ID**

Required argument

## details

Get room details

```
synadm room details [OPTIONS] ROOM_ID
```

## Arguments

### ROOM\_ID

Required argument

## join

Join a room

```
synadm room join [OPTIONS] ROOM_ID_OR_ALIAS USER_ID
```

## Arguments

### ROOM\_ID\_OR\_ALIAS

Required argument

### USER\_ID

Required argument

## list

List and search for rooms.

```
synadm room list [OPTIONS]
```

## Options

**-f, --from** <from>

Offset room listing by given number. This option is used for pagination.

### Default

0

**-l, --limit** <limit>

Maximum amount of rooms to return.

### Default

100

**-n, --name** <name>

Filter rooms by parts of their room name, canonical alias and room id.

**-s, --sort** <sort>

The method in which to sort the returned list of rooms.

### Options

name | canonical\_alias | joined\_members | joined\_local\_members | version | creator | encryption  
| federatable | public | join\_rules | guest\_access | history\_visibility | state\_events

**-r, --reverse**

Direction of room order. If set it will reverse the sort order of --order-by method.



## make-admin

Grant a user room admin permission.

If the user is not in the room, and it is not publicly joinable, then invite the user.

```
synadm room make-admin [OPTIONS] ROOM_ID
```

## Options

**-u, --user-id** <user\_id>

By default the server admin (the caller) is granted power, but another user can optionally be specified.

## Arguments

**ROOM\_ID**

Required argument

## members

List current room members

```
synadm room members [OPTIONS] ROOM_ID
```

## Arguments

**ROOM\_ID**

Required argument

## power-levels

List user's power levels set in rooms.

A combination of commands *room list* and *room state*. It enriches the room list response with a list of users and their corresponding power levels set. It only displays a subset of the available information (room name, id, aliases and power levels). Increase the number of rooms fetched using `--limit/-l` (default: 10) or use the pagination option `--from/-f` to go beyond the default. Use `--name/-n` to search. This command can require quite some time to complete depending on those options.

```
synadm room power-levels [OPTIONS]
```

## Options

**-i, --room-id** <room\_id>

View power levels of this room only.

**-a, --all-details**

Show detailed information about each room. The default is to only show room\_id, name, canonical\_alias and power\_levels.

**-f, --from** <from\_>

Offset room listing by given number. This option is used for pagination.

**Default**

0

**-l, --limit** <limit>

Maximum amount of rooms to return.

**Default**

10

**-n, --name** <name>

Filter rooms by parts of their room name, canonical alias and room id.

**-s, --sort** <sort>

The method in which to sort the returned list of rooms.

**Options**

name | canonical\_alias | joined\_members | joined\_local\_members | version | creator | encryption  
| federatable | public | join\_rules | guest\_access | history\_visibility | state\_events

**-r, --reverse**

Direction of room order. If set it will reverse the sort order of --order-by method.

## resolve

Lookup room ID from alias or vice versa

```
synadm room resolve [OPTIONS] ROOM_ID_OR_ALIAS
```

## Options

**-r, --reverse**

Fetch all room aliases corresponding to a given room ID, instead of the other way round.

**Default**

False

## Arguments

### ROOM\_ID\_OR\_ALIAS

Required argument

## search

An alias to *synadm room list -n <search-term>*.

```
synadm room search [OPTIONS] SEARCH_TERM
```

## Options

**-f, --from** <from\_>

Offset room listing by given number. This option is used for pagination.

### Default

0

**-l, --limit** <limit>

Maximum amount of rooms to return.

### Default

100

**-s, --sort** <sort>

The method in which to sort the returned list of rooms.

### Options

name | canonical\_alias | joined\_members | joined\_local\_members | version | creator | encryption  
| federatable | public | join\_rules | guest\_access | history\_visibility | state\_events

**-r, --reverse**

Direction of room order. If set it will reverse the sort order of `--order-by` method.

## Arguments

### SEARCH\_TERM

Required argument

## state

Get a list of all state events in a room.

```
synadm room state [OPTIONS] ROOM_ID
```

## Arguments

### ROOM\_ID

Required argument

## 1.5 Media

### 1.5.1 synadm media

Manage local and remote media

```
synadm media [OPTIONS] COMMAND [ARGS]...
```

#### delete

Delete media by ID, size or age

```
synadm media delete [OPTIONS]
```

## Options

**-i, --media-id** <media\_id>

The media with this specific media ID will be deleted.

**-d, --before-days** <before\_days>

Delete all media that was last accessed before this number of days ago.

**-b, --before** <before>

Delete all media that was last accessed before this date/time. Eg. '2021-01-01', see above for available date/time formats.

**-t, --before-ts** <before\_ts>

Delete all media that was last accessed before this unix timestamp in ms.

**--size, --kib** <size>

Delete all media that is larger than this size in KiB (1 KiB = 1024 bytes).

**--delete-profiles, --all**

Also delete files that are still used in image data (e.g user profile, room avatar). If set, these files will be deleted too. Not valid when a specific media is being deleted (`-media-id`)

## list

List local media by room or user

```
synadm media list [OPTIONS]
```

### Options

**-r, --room-id** <room\_id>

List all media in room with this room ID ('!abcdefg').

**-u, --user-id** <user\_id>

List all media uploaded by user with this matrix ID (@user:server).

**-f, --from** <from\_>

Offset media listing by given number. This option is also used for pagination but only supported together with `-user-id`.

#### Default

0

**-l, --limit** <limit>

Limit media listing to given number. This option is only supported together with `-user-id`.

#### Default

100

**-s, --sort** <sort>

The method by which to sort the returned list of media. If the ordered field has duplicates, the second order is always by ascending `media_id`, which guarantees a stable ordering. This option is only supported together with `-user-id`.

#### Options

`media_id` | `upload_name` | `created_ts` | `last_access_ts` | `media_length` | `media_type` | `quarantined_by` | `safe_from_quarantine`

**-R, --reverse**

Direction of media order. If set it will reverse the sort order of `-order-by` method. This option is only supported together with `-user-id`.

**--datetime, --dt, --timestamp, --ts**

Display created and last accessed timestamps in a human readable format, or as a unix timestamp in milliseconds. This option only applies to user media and is ignored with room media. [default: datetime].

## protect

Protect specific media from being quarantined

```
synadm media protect [OPTIONS] MEDIA_ID
```

## Arguments

### MEDIA\_ID

Required argument

## purge

Purge old cached remote media

```
synadm media purge [OPTIONS]
```

## Options

**-d, --before-days** <before\_days>

Purge all media that was last accessed before this number of days ago.

**-b, --before** <before>

Purge all media that was last accessed before this date/time. Eg. '2021-01-01', see above for available date/time formats.

**-t, --before-ts** <before\_ts>

Purge all media that was last accessed before this unix timestamp in ms.

## quarantine

Quarantine media in rooms, by users or by media ID

```
synadm media quarantine [OPTIONS]
```

## Options

**-i, --media-id** <media\_id>

The media with this specific media ID will be quarantined.

**-r, --room-id** <room\_id>

All media in room with this room ID (!abcdefg) will be quarantined.

**-u, --user-id** <user\_id>

All media uploaded by user with this matrix ID (@user:server) will be quarantined.

**-s, --server-name** <server\_name>

The server name of the media, mandatory when `-media-id` is used and `_remote_` media should be processed. For locally stored media this option can be omitted.

## 1.6 History

### 1.6.1 synadm history

Purge historic events from Synapse database

```
synadm history [OPTIONS] COMMAND [ARGS]...
```

#### purge

Purge room events before a point in time or before an event ID.

The purge history API allows server admins to purge historic events from their database, reclaiming disk space.

Depending on the amount of history being purged a call to the API may take several minutes or longer. During this period users will not be able to paginate further back in the room from the point being purged from.

Note that Synapse requires at least one message in each room, so it will never delete the last message in a room.

By default, events sent by local users are not deleted, as they may represent the only copies of this content in existence. (Events sent by remote users are deleted.)

Room state data (such as joins, leaves, topic) is always preserved.

The API starts the purge running, and returns immediately with a JSON body with a purge id. Use 'synadm history purge-status <purge id>' to poll for updates on the running purge.

```
synadm history purge [OPTIONS] ROOM_ID
```

#### Options

**-i, --before-event-id** <before\_event\_id>

Purge all history before this event ID. Event ID's contain \$, which is interpreted by the shell. Make sure it is surrounded by single quotes ('id').

**-d, --before-days** <before\_days>

Purge all history before this number of days ago.

**--before** <before>

Purge all history before this point in time. Eg. '2021-01-01', see above for available date/time formats.

**--before-ts** <before\_ts>

Purge all history before this point in time giving a unix timestamp in ms.

**--delete-local**

This option overrides the default behavior and forces removal of events sent by local users.

## Arguments

### ROOM\_ID

Required argument

## purge-status

View status of a recent history purge. Provide purge ID as argument.

The status will be one of active, complete, or failed.

```
synadm history purge-status [OPTIONS] PURGE_ID
```

## Arguments

### PURGE\_ID

Required argument

## 1.7 Group

### 1.7.1 synadm group

Manage groups (communities)

```
synadm group [OPTIONS] COMMAND [ARGS]...
```

## delete

Delete a local group (community)

```
synadm group delete [OPTIONS] GROUP_ID
```

## Arguments

### GROUP\_ID

Required argument

## 1.8 Matrix

### 1.8.1 synadm matrix

Execute Matrix API calls.

```
synadm matrix [OPTIONS] COMMAND [ARGS]...
```



## login

Login to Matrix via username/password and receive an access token.

The response also contains a newly generated device ID and further information about user and homeserver.

Each successful login will show up in the user's devices list marked with a display name of 'synadm matrix login command'.

```
synadm matrix login [OPTIONS] USER_ID
```

## Options

**-p, --password** <password>

The Matrix user's password. If missing, an interactive password prompt is shown.

## Arguments

**USER\_ID**

Required argument

## raw

Execute a raw request to the Matrix API.

The endpoint argument is the part of the URL *after* the configured base URL and Matrix path (see *synadm config*). A simple get request would e.g look like this: *synadm matrix raw client/versions*

Use either *-token* or *-prompt* to provide a user's token and execute Matrix commands on their behalf. Respect the privacy of others! Be responsible!

The precedence rules for token reading are: 1. Interactive input using *-prompt*; 2. Set on CLI via *-token* string; 3. Read from environment variable \$MTOKEN; 4. Preconfigured admin token set in synadm's config file.

```
synadm matrix raw [OPTIONS] ENDPOINT
```

## Options

**-m, --method** <method>

The HTTP method used for the request.

**Default**

get

**Options**

get | post | put | delete

**-d, --data** <data>

The JSON string sent in the body of post, put and delete requests - provided as a string. Make sure to escape it from shell interpretation by using single quotes. E.g '{ "key1": "value1", "key2": 123 }'

**Default**

{ }

**-f, --data-file** <data\_file>

Read JSON data from file. To read from stdin use “-” as the filename argument.

**-t, --token** <token>

Token used for Matrix authentication instead of the configured admin user’s token. If `-token` (and `-prompt`) option is missing, the token is read from environment variable `$MTOKEN` instead. To make sure a user’s token does not show up in system logs, don’t provide it on the shell directly but set `$MTOKEN` with shell command *read MTOKEN*.

**-p, --prompt**

Prompt for the token used for Matrix authentication. This option always overrides `$MTOKEN`.

**Default**

False

## Arguments

### ENDPOINT

Required argument

## Environment variables

### MTOKEN

Provide a default for `--token`

# 1.9 Regtok

## 1.9.1 synadm regtok

Manage registration tokens

```
synadm regtok [OPTIONS] COMMAND [ARGS]...
```

### delete

Delete a registration token

```
synadm regtok delete [OPTIONS] TOKEN
```

## Arguments

### TOKEN

Required argument

## details

View details of the given token

```
synadm regtok details [OPTIONS] TOKEN
```

## Options

**--datetime, --dt, --timestamp, --ts**

Display expiry time in a human readable format, or as a unix timestamp in milliseconds. [default: datetime].

## Arguments

**TOKEN**

Required argument

## list

List registration tokens

```
synadm regtok list [OPTIONS]
```

## Options

**-v, --valid, -V, --invalid**

List only valid/invalid tokens.

**--datetime, --dt, --timestamp, --ts**

Display expiry time in a human readable format, or as a unix timestamp in milliseconds. [default: datetime].

## new

Create a new registration token

```
synadm regtok new [OPTIONS]
```

## Options

**-n, --token <token>**

Set the registration token. The default is a random value generated by the server.

**-l, --length <length>**

The length of the randomly generated token if the token is not specified.

**Default**

16

**-u, --uses-allowed** <uses\_allowed>

The number of times the token can be used to complete a registration before it becomes invalid. [default: unlimited uses]

**-t, --expiry-ts** <expiry\_ts>

The latest time the registration token is valid. Given as the number of milliseconds since 1970-01-01 00:00:00 UTC.

[default: no expiry]

**-e, --expire-at** <expire\_at>

The latest time the registration token is valid. See above for available date/time formats. [default: no expiry]

## update

Update a registration token

```
synadm regtok update [OPTIONS] TOKEN
```

## Options

**-u, --uses-allowed** <uses\_allowed>

The number of times the token can be used to complete a registration before it becomes invalid. Use -1 for an unlimited number of uses. [default: unchanged]

**-t, --expiry-ts** <expiry\_ts>

The latest time the registration token is valid. Given as the number of milliseconds since 1970-01-01 00:00:00 UTC. Use -1 for no expiration. [default: unchanged]

**-e, --expire-at** <expire\_at>

The latest time the registration token is valid. See above for available date/time formats. [default: unchanged]

## Arguments

**TOKEN**

Required argument

## PYTHON PACKAGE DOCUMENTATION

### 2.1 Synadm CLI Package

The `synadm.cli` package contains all the command line frontend code (using the Python Click module), see the sections for each subcommand for help on usage and have a look into the files inside the `synadm/cli` directory to see how the CLI frontend is constructed. The main command is defined in `__init__.py` and each subcommand lives in its own module (e.g `user.py`, `room.py`, ...)

Click command definition code is not very well suited for autogenerated Python package documentations (in this case using Sphinx autodoc) but one thing in this package that is worth mentioning and generating documentation for, is the [`synadm.cli.APIHelper`](#) class which is the connection between the CLI code and the backend code in [`synadm.api`](#):

CLI base functions and settings

**class** `synadm.cli.APIHelper`(*config\_path, verbose, batch, output\_format\_cli*)

Bases: `object`

API client enriched with CLI-level functions, used as a proxy to the client object.

```
FORMATTERS = {'human': <function humanize>, 'json': <function dumps>, 'pprint':  
<function pformat>, 'yaml': <function dump>}
```

```
CONFIG = {'admin_path': '/_synapse/admin', 'base_url': 'http://localhost:8008',  
'homeserver': 'auto-retrieval', 'matrix_path': '/_matrix', 'server_discovery':  
'well-known', 'timeout': 30, 'token': '', 'user': ''}
```

**init\_logger**(*verbose*)

Log both to console (defaults to WARNING) and file (DEBUG).

**\_set\_formatter**(*\_output\_format*)

**load**()

Load the configuration and initialize the client.

**write\_config**(*config*)

Write a new version of the configuration to file.

**output**(*data*)

Output data object using the configured formatter.

**retrieve\_homeserver\_name**(*uri=None*)

Try to retrieve the homeserver name.

When homeserver is set in the config already, it's just returned and nothing is tried to be fetched automatically. If not, either the location of the Federation API is looked up via a `.well-known` resource or a DNS

SRV lookup. This depends on the `server_discovery` setting in the config. Finally the Federation API is used to retrieve the homeserver name.

**Parameters**

**uri** (*string*) – proto://name:port or proto://fqdn:port

**Returns**

hostname, FQDN or DOMAIN; or None on errors.

**Return type**

string

**generate\_mxid(*user\_id*)**

Checks whether the given user ID is an MXID already or else generates it from the passed string and the homeserver name fetched via the `retrieve_homeserver_name` method.

**Parameters**

**user\_id** (*string*) – User ID given by user as command argument.

**Returns**

**the fully qualified Matrix User ID (MXID) or None if the**  
user\_id parameter is None.

**Return type**

string

## 2.2 Synadm Package

The `synadm` package contains the `synadm.api` module which is responsible for accessing the Synapse admin API endpoints and the regular Matrix API endpoints. Also some helper methods, e.g to calculate time-related things are to be found in this module:

Synapse admin API and regular Matrix API clients

Most API calls defined in this module respect the API's defaults and only pass what's necessary in the request body.

A fully qualified Matrix user ID looks like this: `@user:server`, where `server` often is a domain name only, e.g `@user@example.org`

See [https://github.com/matrix-org/synapse/tree/master/docs/admin\\_api](https://github.com/matrix-org/synapse/tree/master/docs/admin_api) for documentation of the Synapse admin APIs and the Matrix spec at <https://matrix.org/docs/spec/#matrix-apis>.

**class** `synadm.api.ApiRequest`(*log, user, token, base\_url, path, timeout, debug*)

Bases: `object`

Basic API request handling and helper utilities

This is subclassed by `SynapseAdmin` and `Matrix`

**query**(*method, urlpart, params=None, data=None, token=None, base\_url\_override=None, verify=True*)

Generic wrapper around requests methods.

Handles requests methods, logging and exceptions.

**Parameters**

- **urlpart** (*string*) – The path to the API endpoint, excluding `self.base_url` and `self.path` (the part after `proto://fqdn:port/path`).
- **params** (*dict, optional*) – URL parameters (`?param1&paarm2`). Defaults to `None`.

- **data** (*dict*, *optional*) – Request body used in POST, PUT, DELETE requests. Defaults to None.
- **base\_url\_override** (*bool*) – The default setting of self.base\_url set on initialization can be overwritten using this argument.
- **verify** (*bool*) – Mandatory SSL verification is turned on by default and can be turned off using this method.

**Returns****Usually a JSON string containing**

the response of the API; responses that are not 200(OK) (usually error messages returned by the API) will also be returned as JSON strings. On exceptions the error type and description are logged and None is returned.

**Return type**

string or None

**`_timestamp_from_days_ago(days)`**

Get a unix timestamp in ms from days ago

**Parameters**

**days** (*int*) – number of days

**Returns**

a unix timestamp in milliseconds (ms)

**Return type**

int

**`_timestamp_from_days_ahead(days)`**

Get a unix timestamp in ms for the given number of days ahead

**Parameters**

**days** (*int*) – number of days

**Returns**

a unix timestamp in milliseconds (ms)

**Return type**

int

**`_timestamp_from_datetime(_datetime)`**

Get a unix timestamp in ms from a datetime object

**Parameters**

**\_datetime** (*datetime object*) – an object built by datetime.datetime

**Returns**

a unix timestamp in milliseconds (ms)

**Return type**

int

**`_datetime_from_timestamp(timestamp, as_str=False)`**

Get a datetime object from a unix timestamp in ms

**Parameters**

**timestamp** (*int*) – a unix timestamp in milliseconds (ms)

**Returns**

**an object built by `datetime.datetime`.**

If `as_str` is set, return a string formatted by `self._format_datetime()` instead.

**Return type**

`datetime object`

**`_format_datetime(datetime_obj)`**

Get a formatted date as a string.

**Parameters**

**`datetime_obj`** (*int*) – A `datetime` object.

**Returns**

**A date in the format we use it throughout synadm. No sanity checking.**

**Return type**

`string`

**class** `synadm.api.MiscRequest`(*log, timeout, debug*)

Bases: [`ApiRequest`](#)

Miscellaneous HTTP requests

**Inheritance:**

**`ApiRequest` (object): parent class containing general properties and methods for requesting REST API's**

**`federation_uri_well_known(base_url)`**

Retrieve the URI to the Server-Server (Federation) API port via the `.well-known` resource of a Matrix server/domain.

**Parameters**

**`base_url`** – `proto://name` or `proto://fqdn`

**Returns**

**`https://fqdn:port` of the delegated server for Server-Server communication between Matrix homeservers or `None` on errors.**

**Return type**

`string`

**class** `synadm.api.Matrix`(*log, user, token, base\_url, matrix\_path, timeout, debug*)

Bases: [`ApiRequest`](#)

Matrix API client

**Inheritance:**

**`ApiRequest` (object): parent class containing general properties and methods for requesting REST API's**

**`user_login(user_id, password)`**

Login as a Matrix user and retrieve an access token

**Parameters**

- **`user_id`** (*string*) – a fully qualified Matrix user ID
- **`password`** (*string*) – the Matrix user's password

**Returns**



**JSON string containing a token suitable to access the**

Matrix API on the user's behalf, a device\_id and some more details on Matrix server and user.

**Return type**

string

**room\_get\_id**(*room\_alias*)

Get the room ID for a given room alias

**Parameters**

**room\_alias** (*string*) – A Matrix room alias (#name:example.org)

**Returns**

**A dict containing the room ID for the alias.**

If room\_id is missing in the response we return the whole response as it might contain Synapse's error message.

**Return type**

string, dict or None

**room\_get\_aliases**(*room\_id*)

Get a list of room aliases for a given room ID

**Parameters**

**room\_id** (*string*) – A Matrix room ID (!abc123:example.org)

**Returns**

**A dict containing a list of room aliases, Synapse's**

error message or None on exceptions.

**Return type**

dict or None

**raw\_request**(*endpoint, method, data, token=None*)

**server\_name\_keys\_api**(*server\_server\_uri*)

Retrieve the Matrix server's own homeserver name via the Server-Server (Federation) API.

**Parameters**

**server\_server\_uri** (*string*) – proto://name:port or proto://fqdn:port

**Returns**

The Matrix server's homeserver name or FQDN, usually something like matrix.DOMAIN or DOMAIN

**Return type**

string

**class** synadm.api.**SynapseAdmin**(*log, user, token, base\_url, admin\_path, timeout, debug*)

Bases: [ApiRequest](#)

Synapse admin API client

**Inheritance:**

**ApiRequest (object):** parent class containing general properties and methods for requesting REST API's

**user\_list**(*\_from*, *\_limit*, *\_guests*, *\_deactivated*, *\_name*, *\_user\_id*)

List and search users

**Parameters**

- **\_from** (*int*) – offsets user list by this number, used for pagination
- **\_limit** (*int*) – maximum number of users returned, used for pagination
- **\_guests** (*bool*) – enable/disable fetching of guest users
- **\_deactivated** (*bool*) – enable/disable fetching of deactivated users
- **\_name** (*string*) – user name localpart to search for, see Synapse admin API docs for details
- **\_user\_id** (*string*) – fully qualified Matrix user ID to search for

**Returns**

JSON string containing the found users

**Return type**

string

**user\_membership**(*user\_id*, *return\_aliases*, *matrix\_api*)

Get a list of rooms the given user is member of

**Parameters**

- **user\_id** (*string*) – Fully qualified Matrix user ID
- **room\_aliases** (*bool*) – Return human readable room aliases instead of room ID's if applicable.
- **matrix\_api** (*object*) – An initialized Matrix object needs to be passes as we need some Matrix API functionality here.

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**user\_deactivate**(*user\_id*, *gdpr\_erase*)

Delete a given user

**Parameters**

- **user\_id** (*string*) – fully qualified Matrix user ID
- **gdpr\_erase** (*bool*) – enable/disable gdpr-erasing the user, see Synapse admin API docs for details.

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**user\_password**(*user\_id*, *password*, *no\_logout*)

Set the user password, and log the user out if requested

**Parameters**

- **user\_id** (*string*) – fully qualified Matrix user ID
- **password** (*string*) – new password that should be set
- **no\_logout** (*bool*) – the API defaults to logging out the user after password reset via the admin API, this option can be used to disable this behaviour.

**Returns**

JSON string containing the admin API's response or None if an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**user\_details**(*user\_id*)

Get information about a given user

Note that the admin API docs describe this function as “Query User Account”.

**Parameters**

**user\_id** (*string*) – fully qualified Matrix user ID

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**user\_login**(*user\_id*, *expire\_days*, *expire*, *\_expire\_ts*)

Get an access token that can be used to authenticate as that user.

If one of the args *expire\_days*, *expire* or *\_expire\_ts* is set, the *valid\_until\_ms* field will be sent to the API endpoint. If this is not the case the default of the API would be used. At the time of writing, this would be that tokens never expire.

Note: If this method is called by the CLI frontend code (`synadm.cli.user.user_login_cmd`), a default expiry date of 1 day (24h) is passed.

**Parameters**

- **user\_id** (*string*) – fully qualified Matrix user ID
- **expire\_days** (*int*) – token should expire after this number of days
- **expire** (*datetime*) – token should expire after this date/time - a datetime object (e.g. as generated by `Click.DateTime()`)
- **\_expire\_ts** (*int*) – token should expire after this date/time - a unix timestamp in ms.

**Returns**

JSON string containing the admin API's response or None if an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**user\_modify**(*user\_id*, *password*, *display\_name*, *threepid*, *avatar\_url*, *admin*, *deactivation*)

Create or update information about a given user

Threepid should be passed as a tuple in a tuple

**user\_whois**(*user\_id*)

Return information about the active sessions for a specific user

**user\_devices**(*user\_id*)

Return information about all devices for a specific user.

**Parameters**

**user\_id** (*string*) – Fully qualified Matrix user ID.

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**user\_devices\_get\_todelete**(*devices\_data*, *min\_days*, *min\_surviving*, *device\_id*, *readable\_seen*)

Gather a list of devices that possibly could be deleted.

This method is used by the 'user prune-devices' command.

**Parameters**

- **devices\_data** (*list*) – Containing dicts of all the user's devices, as returned by the user\_devices method (the user/devices API endpoint).
- **min\_days** (*int*) – At least this number of days need to have passed from the last time a device was seen for it to be deleted. A reasonable default should be sent by the CLI level method.
- **min\_surviving** – At least this amount of devices will be kept alive. A reasonable default should be sent by the CLI level method.
- **device\_id** – Only search devices with this ID.
- **datetime** – When True, 'last seen timestamp' is replaced with a human readable format.

**Returns**

**Containing dicts of devices that possibly could be deleted.**  
If non apply, an empty list is returned.

**Return type**

list

**user\_devices\_delete**(*user\_id*, *devices*)

Delete the specified devices for a specific user. Returns an empty JSON dict.

devices is a list of device IDs

**room\_join**(*room\_id\_or\_alias*, *user\_id*)

Allow an administrator to join an user account with a given user\_id to a room with a given room\_id\_or\_alias

**room\_list**(*\_from*, *limit*, *name*, *order\_by*, *reverse*)

List and search rooms

**room\_details**(*room\_id*)

Get details about a room

**room\_members**(*room\_id*)

Get a list of room members

**room\_state(room\_id)**

Get a list of all state events in a room.

**Parameters**

**room\_id** (*string*) –

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**room\_power\_levels**(*from\_, limit, name, order\_by, reverse, room\_id=None, all\_details=True, output\_format='json'*)

Get a list of configured power\_levels in all rooms.  
or a single room.

**Parameters**

**room\_id** (*string*) – If left out, all rooms are fetched.

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**room\_delete**(*room\_id, new\_room\_user\_id, room\_name, message, block, no\_purge*)

Delete a room and purge it if requested

**room\_make\_admin**(*room\_id, user\_id*)

Grant a user room admin permission. If the user is not in the room, and it is not publicly joinable, then invite the user.

**room\_media\_list**(*room\_id*)

Get a list of known media in an (unencrypted) room.

**media\_quarantine**(*server\_name, media\_id*)

Quarantine a single piece of local or remote media

**room\_media\_quarantine**(*room\_id*)

Quarantine all local and remote media in a room

**user\_media\_quarantine**(*user\_id*)

Quarantine all local and remote media of a user

**user\_media**(*user\_id, \_from, limit, order\_by, reverse, readable*)

Get a user's uploaded media

**media\_delete**(*server\_name, media\_id*)

Delete a specific (local) media\_id

**media\_delete\_by\_date\_or\_size**(*server\_name, before\_days, before, \_before\_ts, \_size\_gt, delete\_profiles*)

Delete local media by date and/or size FIXME and/or?

**media\_protect**(*media\_id*)

Protect a single piece of local or remote media  
from being quarantined

**purge\_media\_cache**(*before\_days*, *before*, *\_before\_ts*)

Purge old cached remote media

**version**()

Get the server version

**group\_delete**(*group\_id*)

Delete a local group (community)

**purge\_history**(*room\_id*, *before\_event\_id*, *before\_days*, *before*, *\_before\_ts*, *delete\_local*)

Purge room history

**purge\_history\_status**(*purge\_id*)

Get status of a recent history purge  
The status will be one of active, complete, or failed.

**regtok\_list**(*valid*, *readable\_expiry*)

List registration tokens

**Parameters**

- **valid** (*bool*) – List only valid (if True) or invalid (if False) tokens. Default is to list all tokens regardless of validity.
- **readable\_expiry** (*bool*) – If True, replace the `expiry_time` field with a human readable datetime. If False, `expiry_time` will be a unix timestamp.

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**regtok\_details**(*token*, *readable\_expiry*)

Get details about the given registration token

**Parameters**

- **token** (*string*) – The registration token in question
- **readable\_expiry** (*bool*) – If True, replace the `expiry_time` field with a human readable datetime. If False, `expiry_time` will be a unix timestamp.

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**regtok\_new**(*token*, *length*, *uses\_allowed*, *expiry\_ts*, *expire\_at*)

Create a new registration token

**Parameters**

- **token** (*string*) – Registration token to create. Default is randomly generated by the server.
- **length** (*int*) – The length of the token to generate if the token is not provided.
- **uses\_allowed** (*int*) – The number of times the token can be used to complete a registration before it becomes invalid.
- **expiry\_ts** (*int*) – The latest time the registration token is valid. Given as the number of milliseconds since 1970-01-01 00:00:00 UTC.
- **expire\_at** (*click.DateTime*) – The latest time the registration token is valid.

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**regtok\_update**(*token, uses\_allowed, expiry\_ts, expire\_at*)

Update a registration token

**Parameters**

- **token** (*string*) – Registration token to update.
- **uses\_allowed** (*int*) – The number of times the token can be used to complete a registration before it becomes invalid.
- **expiry\_ts** (*int*) – The latest time the registration token is valid. Given as the number of milliseconds since 1970-01-01 00:00:00 UTC. -1 indicates no expiry.
- **expire\_at** (*click.DateTime*) – The latest time the registration token is valid.

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**regtok\_delete**(*token*)

Delete a registration token

**Parameters**

- **token** (*string*) – The registration token to delete

**Returns**

**JSON string containing the admin API's response or None if**  
an exception occurred. See Synapse admin API docs for details.

**Return type**

string

**user\_shadow\_ban**(*user\_id, unban*)

Shadow-ban or unban a user.

**Parameters**

- **user\_id** (*string*) – The user to be banned/unbanned.
- **unban** (*boolean*) – Unban the specified user.





## INDICES AND TABLES

- `genindex`
- `modindex`
- `search`



## PYTHON MODULE INDEX

### S

`synadm.api`, [26](#)

`synadm.cli`, [25](#)



## Symbols

- `_datetime_from_timestamp()`  
(*synadm.api.ApiRequest* method), 27
- `_format_datetime()` (*synadm.api.ApiRequest* method), 28
- `_set_formatter()` (*synadm.cli.APIHelper* method), 25
- `_timestamp_from_datetime()`  
(*synadm.api.ApiRequest* method), 27
- `_timestamp_from_days_ago()`  
(*synadm.api.ApiRequest* method), 27
- `_timestamp_from_days_ahead()`  
(*synadm.api.ApiRequest* method), 27
- G  
synadm-user-list command line option, 4
- P  
synadm-user-modify command line option, 7
- R  
synadm-media-list command line option, 17
- V  
synadm-regtok-list command line option, 23
- activate  
synadm-user-modify command line option, 7
- admin  
synadm-user-modify command line option, 7
- admin-path  
synadm-config command line option, 2
- aliases  
synadm-user-membership command line option, 6
- all  
synadm-media-delete command line option, 16
- all-details  
synadm-room-power-levels command line option, 14
- avatar-url  
synadm-user-modify command line option, 7
- base-url  
synadm-config command line option, 2
- batch  
synadm command line option, 1
- before  
synadm-history-purge command line option, 19
- synadm-media-delete command line option, 16
- synadm-media-purge command line option, 18
- before-days  
synadm-history-purge command line option, 19
- synadm-media-delete command line option, 16
- synadm-media-purge command line option, 18
- before-event-id  
synadm-history-purge command line option, 19
- before-ts  
synadm-history-purge command line option, 19
- synadm-media-delete command line option, 16
- synadm-media-purge command line option, 18
- block  
synadm-room-delete command line option, 11
- config-file  
synadm command line option, 1
- data  
synadm-matrix-raw command line option, 21
- data-file  
synadm-matrix-raw command line option, 21
- datetime  
synadm-media-list command line option, 17
- synadm-regtok-details command line option, 23
- synadm-regtok-list command line option, 23
- synadm-user-media command line option, 6
- synadm-user-prune-devices command line option, 9

--deactivate  
    synadm-user-modify command line option, 7

--deactivated  
    synadm-user-list command line option, 4

--delete-local  
    synadm-history-purge command line option, 19

--delete-profiles  
    synadm-media-delete command line option, 16

--device-id  
    synadm-user-prune-devices command line option, 9

--display-name  
    synadm-user-modify command line option, 7

--dt  
    synadm-media-list command line option, 17  
    synadm-regtok-details command line option, 23  
    synadm-regtok-list command line option, 23  
    synadm-user-media command line option, 6  
    synadm-user-prune-devices command line option, 9

--expire  
    synadm-user-login command line option, 5

--expire-at  
    synadm-regtok-new command line option, 24  
    synadm-regtok-update command line option, 24

--expire-days  
    synadm-user-login command line option, 5

--expire-never  
    synadm-user-login command line option, 5

--expire-ts  
    synadm-user-login command line option, 5

--expiry-ts  
    synadm-regtok-new command line option, 24  
    synadm-regtok-update command line option, 24

--from  
    synadm-media-list command line option, 17  
    synadm-room-list command line option, 12  
    synadm-room-power-levels command line option, 14  
    synadm-room-search command line option, 15  
    synadm-user-list command line option, 4  
    synadm-user-media command line option, 6  
    synadm-user-search command line option, 9

--gdpr-erase  
    synadm-user-deactivate command line option, 3

--guests  
    synadm-user-list command line option, 4

--homeserver  
    synadm-config command line option, 3

--ids  
    synadm-user-membership command line option, 6

--invalid  
    synadm-regtok-list command line option, 23

--kib  
    synadm-media-delete command line option, 16

--length  
    synadm-regtok-new command line option, 23

--limit  
    synadm-media-list command line option, 17  
    synadm-room-list command line option, 12  
    synadm-room-power-levels command line option, 14  
    synadm-room-search command line option, 15  
    synadm-user-list command line option, 4  
    synadm-user-media command line option, 6  
    synadm-user-search command line option, 9

--list-only  
    synadm-user-prune-devices command line option, 8

--matrix-path  
    synadm-config command line option, 2

--media-id  
    synadm-media-delete command line option, 16  
    synadm-media-quarantine command line option, 18

--message  
    synadm-room-delete command line option, 11

--method  
    synadm-matrix-raw command line option, 21

--min-days  
    synadm-user-prune-devices command line option, 8

--min-surviving  
    synadm-user-prune-devices command line option, 8

--name  
    synadm-room-list command line option, 12  
    synadm-room-power-levels command line option, 14  
    synadm-user-list command line option, 4

--new-room-user-id  
    synadm-room-delete command line option, 11

--no-admin

---

```

    synadm-user-modify command line option, 7
--no-batch
    synadm command line option, 1
--no-guests
    synadm-user-list command line option, 4
--no-logout
    synadm-user-password command line
        option, 8
--no-purge
    synadm-room-delete command line option,
        11
--output
    synadm command line option, 1
    synadm-config command line option, 2
--password
    synadm-matrix-login command line option,
        21
    synadm-user-modify command line option, 7
    synadm-user-password command line
        option, 8
--password-prompt
    synadm-user-modify command line option, 7
--prompt
    synadm-matrix-row command line option, 22
--reverse
    synadm-media-list command line option, 17
    synadm-room-list command line option, 12
    synadm-room-power-levels command line
        option, 14
    synadm-room-resolve command line option,
        14
    synadm-room-search command line option,
        15
    synadm-user-media command line option, 6
--room-id
    synadm-media-list command line option, 17
    synadm-media-quarantine command line
        option, 18
    synadm-room-power-levels command line
        option, 14
--room-name
    synadm-room-delete command line option,
        11
--server-discovery
    synadm-config command line option, 3
--server-name
    synadm-media-quarantine command line
        option, 18
--size
    synadm-media-delete command line option,
        16
--sort
    synadm-media-list command line option, 17
    synadm-room-list command line option, 12
    synadm-room-power-levels command line
        option, 14
    synadm-room-search command line option,
        15
    synadm-user-media command line option, 6
--threepid
    synadm-user-modify command line option, 7
--timeout
    synadm-config command line option, 2
--timestamp
    synadm-media-list command line option, 17
    synadm-regtok-details command line
        option, 23
    synadm-regtok-list command line option,
        23
    synadm-user-media command line option, 6
    synadm-user-prune-devices command line
        option, 9
--token
    synadm-config command line option, 2
    synadm-matrix-row command line option, 22
    synadm-regtok-new command line option, 23
--ts
    synadm-media-list command line option, 17
    synadm-regtok-details command line
        option, 23
    synadm-regtok-list command line option,
        23
    synadm-user-media command line option, 6
    synadm-user-prune-devices command line
        option, 9
--unban
    synadm-user-shadow-ban command line
        option, 10
--user
    synadm-config command line option, 2
--user-id
    synadm-media-list command line option, 17
    synadm-media-quarantine command line
        option, 18
    synadm-room-make-admin command line
        option, 13
    synadm-user-list command line option, 4
--uses-allowed
    synadm-regtok-new command line option, 23
    synadm-regtok-update command line
        option, 24
--valid
    synadm-regtok-list command line option,
        23
--verbose
    synadm command line option, 1
-a

```

- synadm-room-power-levels command line option, 14
- synadm-user-modify command line option, 7
- b
  - synadm-config command line option, 2
  - synadm-media-delete command line option, 16
  - synadm-media-purge command line option, 18
  - synadm-room-delete command line option, 11
- c
  - synadm command line option, 1
- d
  - synadm-config command line option, 3
  - synadm-history-purge command line option, 19
  - synadm-matrix-row command line option, 21
  - synadm-media-delete command line option, 16
  - synadm-media-purge command line option, 18
  - synadm-user-list command line option, 4
  - synadm-user-login command line option, 5
  - synadm-user-prune-devices command line option, 8
- e
  - synadm-regtok-new command line option, 24
  - synadm-regtok-update command line option, 24
  - synadm-user-deactivate command line option, 3
- f
  - synadm-matrix-row command line option, 21
  - synadm-media-list command line option, 17
  - synadm-room-list command line option, 12
  - synadm-room-power-levels command line option, 14
  - synadm-room-search command line option, 15
  - synadm-user-list command line option, 4
  - synadm-user-media command line option, 6
  - synadm-user-search command line option, 9
- g
  - synadm-user-list command line option, 4
- i
  - synadm-history-purge command line option, 19
  - synadm-media-delete command line option, 16
  - synadm-media-quarantine command line option, 18
  - synadm-room-power-levels command line option, 14
- synadm-user-list command line option, 4
- synadm-user-prune-devices command line option, 9
- l
  - synadm-media-list command line option, 17
  - synadm-regtok-new command line option, 23
  - synadm-room-list command line option, 12
  - synadm-room-power-levels command line option, 14
  - synadm-room-search command line option, 15
  - synadm-user-list command line option, 4
  - synadm-user-media command line option, 6
  - synadm-user-prune-devices command line option, 8
  - synadm-user-search command line option, 9
- m
  - synadm-config command line option, 2
  - synadm-matrix-row command line option, 21
  - synadm-room-delete command line option, 11
- n
  - synadm-config command line option, 3
  - synadm-regtok-new command line option, 23
  - synadm-room-delete command line option, 11
  - synadm-room-list command line option, 12
  - synadm-room-power-levels command line option, 14
  - synadm-user-list command line option, 4
  - synadm-user-modify command line option, 7
  - synadm-user-password command line option, 8
- o
  - synadm command line option, 1
  - synadm-config command line option, 2
- p
  - synadm-config command line option, 2
  - synadm-matrix-login command line option, 21
  - synadm-matrix-row command line option, 22
  - synadm-user-modify command line option, 7
  - synadm-user-password command line option, 8
- r
  - synadm-media-list command line option, 17
  - synadm-media-quarantine command line option, 18
  - synadm-room-list command line option, 12
  - synadm-room-power-levels command line option, 14
  - synadm-room-resolve command line option, 14



- synadm-room-search command line option, 15
  - synadm-user-media command line option, 6
  - s
    - synadm-media-list command line option, 17
    - synadm-media-quarantine command line option, 18
    - synadm-room-list command line option, 12
    - synadm-room-power-levels command line option, 14
    - synadm-room-search command line option, 15
    - synadm-user-media command line option, 6
    - synadm-user-prune-devices command line option, 8
  - t
    - synadm-config command line option, 2
    - synadm-matrix-raw command line option, 22
    - synadm-media-delete command line option, 16
    - synadm-media-purge command line option, 18
    - synadm-regtok-new command line option, 24
    - synadm-regtok-update command line option, 24
    - synadm-user-modify command line option, 7
  - u
    - synadm-config command line option, 2
    - synadm-media-list command line option, 17
    - synadm-media-quarantine command line option, 18
    - synadm-regtok-new command line option, 23
    - synadm-regtok-update command line option, 24
    - synadm-room-delete command line option, 11
    - synadm-room-make-admin command line option, 13
    - synadm-user-modify command line option, 7
    - synadm-user-shadow-ban command line option, 10
  - v
    - synadm command line option, 1
    - synadm-regtok-list command line option, 23
    - synadm-user-modify command line option, 7
  - w
    - synadm-config command line option, 2
- A**
- APIHelper (*class in synadm.cli*), 25
  - ApiRequest (*class in synadm.api*), 26
- C**
- CONFIG (*synadm.cli.APIHelper attribute*), 25
- E**
- ENDPOINT
    - synadm-matrix-raw command line option, 22
- F**
- federation\_uri\_well\_known() (*synadm.api.MiscRequest method*), 28
  - FORMATTERS (*synadm.cli.APIHelper attribute*), 25
- G**
- generate\_mxid() (*synadm.cli.APIHelper method*), 26
  - group\_delete() (*synadm.api.SynapseAdmin method*), 34
  - GROUP\_ID
    - synadm-group-delete command line option, 20
- I**
- init\_logger() (*synadm.cli.APIHelper method*), 25
- L**
- load() (*synadm.cli.APIHelper method*), 25
- M**
- Matrix (*class in synadm.api*), 28
  - media\_delete() (*synadm.api.SynapseAdmin method*), 33
  - media\_delete\_by\_date\_or\_size() (*synadm.api.SynapseAdmin method*), 33
  - MEDIA\_ID
    - synadm-media-protect command line option, 18
  - media\_protect() (*synadm.api.SynapseAdmin method*), 33
  - media\_quarantine() (*synadm.api.SynapseAdmin method*), 33
  - MiscRequest (*class in synadm.api*), 28
  - module
    - synadm.api, 26
    - synadm.cli, 25
- O**
- output() (*synadm.cli.APIHelper method*), 25
- P**
- purge\_history() (*synadm.api.SynapseAdmin method*), 34
  - purge\_history\_status() (*synadm.api.SynapseAdmin method*), 34

PURGE\_ID

synadm-history-purge-status command  
line option, 20

purge\_media\_cache() (synadm.api.SynapseAdmin  
method), 34

## Q

query() (synadm.api.ApiRequest method), 26

## R

raw\_request() (synadm.api.Matrix method), 29

regtok\_delete() (synadm.api.SynapseAdmin method),  
35

regtok\_details() (synadm.api.SynapseAdmin  
method), 34

regtok\_list() (synadm.api.SynapseAdmin method), 34

regtok\_new() (synadm.api.SynapseAdmin method), 34

regtok\_update() (synadm.api.SynapseAdmin method),  
35

retrieve\_homeserver\_name()  
(synadm.cli.APIHelper method), 25

room\_delete() (synadm.api.SynapseAdmin method), 33

room\_details() (synadm.api.SynapseAdmin method),  
32

room\_get\_aliases() (synadm.api.Matrix method), 29

room\_get\_id() (synadm.api.Matrix method), 29

ROOM\_ID

synadm-history-purge command line  
option, 20

synadm-room-delete command line option,  
11

synadm-room-details command line option,  
12

synadm-room-make-admin command line  
option, 13

synadm-room-members command line option,  
13

synadm-room-state command line option, 16

ROOM\_ID\_OR\_ALIAS

synadm-room-join command line option, 12  
synadm-room-resolve command line option,  
15

room\_join() (synadm.api.SynapseAdmin method), 32  
room\_list() (synadm.api.SynapseAdmin method), 32

room\_make\_admin() (synadm.api.SynapseAdmin  
method), 33

room\_media\_list() (synadm.api.SynapseAdmin  
method), 33

room\_media\_quarantine()  
(synadm.api.SynapseAdmin method), 33

room\_members() (synadm.api.SynapseAdmin method),  
32

room\_power\_levels() (synadm.api.SynapseAdmin  
method), 33

room\_state() (synadm.api.SynapseAdmin method), 32

## S

SEARCH\_TERM

synadm-room-search command line option,  
15

synadm-user-search command line option, 9

server\_name\_keys\_api() (synadm.api.Matrix  
method), 29

synadm command line option

--batch, 1

--config-file, 1

--no-batch, 1

--output, 1

--verbose, 1

-c, 1

-o, 1

-v, 1

synadm.api  
module, 26

synadm.cli  
module, 25

synadm-config command line option

--admin-path, 2

--base-url, 2

--homeserver, 3

--matrix-path, 2

--output, 2

--server-discovery, 3

--timeout, 2

--token, 2

--user, 2

-b, 2

-d, 3

-m, 2

-n, 3

-o, 2

-p, 2

-t, 2

-u, 2

-w, 2

synadm-group-delete command line option  
GROUP\_ID, 20

synadm-history-purge command line option

--before, 19

--before-days, 19

--before-event-id, 19

--before-ts, 19

--delete-local, 19

-d, 19

-i, 19

ROOM\_ID, 20

synadm-history-purge-status command line  
option

---

```

    PURGE_ID, 20
synadm-matrix-login command line option
    --password, 21
    -p, 21
    USER_ID, 21
synadm-matrix-row command line option
    --data, 21
    --data-file, 21
    --method, 21
    --prompt, 22
    --token, 22
    -d, 21
    -f, 21
    -m, 21
    -p, 22
    -t, 22
    ENDPOINT, 22
synadm-media-delete command line option
    --all, 16
    --before, 16
    --before-days, 16
    --before-ts, 16
    --delete-profiles, 16
    --kib, 16
    --media-id, 16
    --size, 16
    -b, 16
    -d, 16
    -i, 16
    -t, 16
synadm-media-list command line option
    -R, 17
    --datetime, 17
    --dt, 17
    --from, 17
    --limit, 17
    --reverse, 17
    --room-id, 17
    --sort, 17
    --timestamp, 17
    --ts, 17
    --user-id, 17
    -f, 17
    -l, 17
    -r, 17
    -s, 17
    -u, 17
synadm-media-protect command line option
    MEDIA_ID, 18
synadm-media-purge command line option
    --before, 18
    --before-days, 18
    --before-ts, 18
    -b, 18
    -d, 18
    -t, 18
synadm-media-quarantine command line option
    --media-id, 18
    --room-id, 18
    --server-name, 18
    --user-id, 18
    -i, 18
    -r, 18
    -s, 18
    -u, 18
synadm-regtok-delete command line option
    TOKEN, 22
synadm-regtok-details command line option
    --datetime, 23
    --dt, 23
    --timestamp, 23
    --ts, 23
    TOKEN, 23
synadm-regtok-list command line option
    -V, 23
    --datetime, 23
    --dt, 23
    --invalid, 23
    --timestamp, 23
    --ts, 23
    --valid, 23
    -v, 23
synadm-regtok-new command line option
    --expire-at, 24
    --expiry-ts, 24
    --length, 23
    --token, 23
    --uses-allowed, 23
    -e, 24
    -l, 23
    -n, 23
    -t, 24
    -u, 23
synadm-regtok-update command line option
    --expire-at, 24
    --expiry-ts, 24
    --uses-allowed, 24
    -e, 24
    -t, 24
    -u, 24
    TOKEN, 24
synadm-room-delete command line option
    --block, 11
    --message, 11
    --new-room-user-id, 11
    --no-purge, 11
    --room-name, 11
    -b, 11

```

-m, 11  
-n, 11  
-u, 11  
ROOM\_ID, 11  
synadm-room-details command line option  
ROOM\_ID, 12  
synadm-room-join command line option  
ROOM\_ID\_OR\_ALIAS, 12  
USER\_ID, 12  
synadm-room-list command line option  
--from, 12  
--limit, 12  
--name, 12  
--reverse, 12  
--sort, 12  
-f, 12  
-l, 12  
-n, 12  
-r, 12  
-s, 12  
synadm-room-make-admin command line option  
--user-id, 13  
-u, 13  
ROOM\_ID, 13  
synadm-room-members command line option  
ROOM\_ID, 13  
synadm-room-power-levels command line option  
--all-details, 14  
--from, 14  
--limit, 14  
--name, 14  
--reverse, 14  
--room-id, 14  
--sort, 14  
-a, 14  
-f, 14  
-i, 14  
-l, 14  
-n, 14  
-r, 14  
-s, 14  
synadm-room-resolve command line option  
--reverse, 14  
-r, 14  
ROOM\_ID\_OR\_ALIAS, 15  
synadm-room-search command line option  
--from, 15  
--limit, 15  
--reverse, 15  
--sort, 15  
-f, 15  
-l, 15  
-r, 15  
-s, 15  
SEARCH\_TERM, 15  
synadm-room-state command line option  
ROOM\_ID, 16  
synadm-user-deactivate command line option  
--gdpr-erase, 3  
-e, 3  
USER\_ID, 4  
synadm-user-details command line option  
USER\_ID, 4  
synadm-user-list command line option  
-G, 4  
--deactivated, 4  
--from, 4  
--guests, 4  
--limit, 4  
--name, 4  
--no-guests, 4  
--user-id, 4  
-d, 4  
-f, 4  
-g, 4  
-i, 4  
-l, 4  
-n, 4  
synadm-user-login command line option  
--expire, 5  
--expire-days, 5  
--expire-never, 5  
--expire-ts, 5  
-d, 5  
USER\_ID, 5  
synadm-user-media command line option  
--datetime, 6  
--dt, 6  
--from, 6  
--limit, 6  
--reverse, 6  
--sort, 6  
--timestamp, 6  
--ts, 6  
-f, 6  
-l, 6  
-r, 6  
-s, 6  
USER\_ID, 6  
synadm-user-membership command line option  
--aliases, 6  
--ids, 6  
USER\_ID, 7  
synadm-user-modify command line option  
-P, 7  
--activate, 7  
--admin, 7

--avatar-url, 7  
 --deactivate, 7  
 --display-name, 7  
 --no-admin, 7  
 --password, 7  
 --password-prompt, 7  
 --threepid, 7  
 -a, 7  
 -n, 7  
 -p, 7  
 -t, 7  
 -u, 7  
 -v, 7  
 USER\_ID, 8  
 synadm-user-password command line option  
 --no-logout, 8  
 --password, 8  
 -n, 8  
 -p, 8  
 USER\_ID, 8  
 synadm-user-prune-devices command line  
 option  
 --datetime, 9  
 --device-id, 9  
 --dt, 9  
 --list-only, 8  
 --min-days, 8  
 --min-surviving, 8  
 --timestamp, 9  
 --ts, 9  
 -d, 8  
 -i, 9  
 -l, 8  
 -s, 8  
 USER\_ID, 9  
 synadm-user-search command line option  
 --from, 9  
 --limit, 9  
 -f, 9  
 -l, 9  
 SEARCH\_TERM, 9  
 synadm-user-shadow-ban command line option  
 --unban, 10  
 -u, 10  
 USER\_ID, 10  
 synadm-user-whois command line option  
 USER\_ID, 10  
 SynapseAdmin (class in synadm.api), 29

## T

TOKEN  
 synadm-regtok-delete command line  
 option, 22

synadm-regtok-details command line  
 option, 23  
 synadm-regtok-update command line  
 option, 24

## U

user\_deactivate() (synadm.api.SynapseAdmin  
 method), 30  
 user\_details() (synadm.api.SynapseAdmin method),  
 31  
 user\_devices() (synadm.api.SynapseAdmin method),  
 32  
 user\_devices\_delete() (synadm.api.SynapseAdmin  
 method), 32  
 user\_devices\_get\_todelete()  
 (synadm.api.SynapseAdmin method), 32  
 USER\_ID  
 synadm-matrix-login command line option,  
 21  
 synadm-room-join command line option, 12  
 synadm-user-deactivate command line  
 option, 4  
 synadm-user-details command line option,  
 4  
 synadm-user-login command line option, 5  
 synadm-user-media command line option, 6  
 synadm-user-membership command line  
 option, 7  
 synadm-user-modify command line option, 8  
 synadm-user-password command line  
 option, 8  
 synadm-user-prune-devices command line  
 option, 9  
 synadm-user-shadow-ban command line  
 option, 10  
 synadm-user-whois command line option, 10  
 user\_list() (synadm.api.SynapseAdmin method), 29  
 user\_login() (synadm.api.Matrix method), 28  
 user\_login() (synadm.api.SynapseAdmin method), 31  
 user\_media() (synadm.api.SynapseAdmin method), 33  
 user\_media\_quarantine()  
 (synadm.api.SynapseAdmin method), 33  
 user\_membership() (synadm.api.SynapseAdmin  
 method), 30  
 user\_modify() (synadm.api.SynapseAdmin method), 31  
 user\_password() (synadm.api.SynapseAdmin method),  
 30  
 user\_shadow\_ban() (synadm.api.SynapseAdmin  
 method), 35  
 user\_whois() (synadm.api.SynapseAdmin method), 31  
 V  
 version() (synadm.api.SynapseAdmin method), 34

## W

`write_config()` (*synadm.cli.APIHelper method*), [25](#)