

Migration from self-hosted to ESS On-Premise

Notes

Migrate from self-hosted to Element Server Suite (On-Premise)

Preparation

This section outlines what you should do ahead of the migration in order to ensure the migration goes as quickly as possible and without issues.

- At the latest 48 hours before your migration is scheduled, set the TTL on any DNS records that need to be updated to the lowest allowed value.
- Upgrade your Synapse to the same version as ESS is running. Generally this will be the latest stable release. https://element.ems.host/_matrix/federation/v1/version is a good indicator.
 - This is not required, but if your Synapse version is not the same as the EMS version, your migration will take longer.
- Check the size of your database:
 - PostgreSQL: Connect to your database and issue the command `\l+`
- Check the size of your media repository and report to your EMS contact.
 - Synapse Media Store: `du -hs /path/to/synapse/media_store/`
 - Matrix Media Repo: <https://github.com/turt2live/matrix-media-repo/blob/master/docs/admin.md#per-server-usage>
- If you are using SQLite instead of PostgreSQL, you should port your database to PostgreSQL by following [this](#) guide before dumping your database

SSH to your matrix server

You might want to run everything in a `tmux` or a `screen` session to avoid disruption in case of a lost SSH connection.

Upgrade Synapse to the same version EES is running

Follow <https://element-hq.github.io/synapse/latest/upgrade.html>

Start Synapse, make sure it's happy Stop Synapse

Create a folder to store everything

```
mkdir -p /tmp/synapse_export  
cd /tmp/synapse_export
```

The guide from here on assumes your current working directory is `/tmp/synapse_export`.

Set restrictive permissions on the folder

If you are working as root: (otherwise set restrictive permissions as needed):

```
chmod 000 /tmp/synapse_export
```

Copy Synapse config

Get the following files :

- Your Synapse configuration file (usually `homeserver.yaml`)
- Your message signing key.
 - This is stored in a separate file. See the Synapse config file [`homeserver.yaml`] for the path. The variable is `signing_key_path` https://element-hq.github.io/synapse/latest/usage/configuration/config_documentation.html#signing_key_path
- grab `macaroon_secret_key` from `homeserver.yaml` and place it in the "Secrets \ Synapse \ Macaroon"

Stop Synapse

DO NOT START IT AGAIN AFTER THIS

Doing so can cause issues with federation and inconsistent data for your users.

While you wait for the database to export or files to transfer, you should edit or create the well-known files and DNS records to point to your new EES host. This can take a while to update so should be done as soon as possible in order to ensure your server will function properly when the migration is complete.

Database export

PostgreSQL

Dump, compress

Replace:

- `<dbhost>` (ip or fqdn for your database server)
- `<dbusername>` (username for your synapse database)
- `<dbname>` (the name of the database for synapse)

```
pg_dump -Fc -O -h <dbhost> -U <dbusername> -d <dbname> -W -f synapse.dump
```

Setup new host

[[LINK TO ON_PREM SETUP DOCS](#)]

Import DB

Enter a bash shell on the Synapse postgres container:

```
kubectl exec -it -n element-onprem synapse-postgres-0 --container postgres -- /bin/bash
```

```
psql -U synapse_user synapse
```

 on postgres container shell

THE FOLLOWING COMMAND WILL ERASE THE EXISTING SYNAPSE DATABASE WITHOUT WARNING OR CONFIRMATION. PLEASE ENSURE THAT IT IS THE CORRECT DB AND THERE IS NO PRODUCTION DATA ON IT

```
DO $$ DECLARE
r RECORD;
BEGIN
  FOR r IN (SELECT tablename FROM pg_tables WHERE schemaname = current_schema()) LOOP
    EXECUTE 'DROP TABLE ' || quote_ident(r.tablename) || ' CASCADE';
  END LOOP;
END $$;

DROP sequence cache_invalidation_stream_seq;
DROP sequence state_group_id_seq;
DROP sequence user_id_seq;
DROP sequence account_data_sequence;
DROP sequence application_services_txn_id_seq;
DROP sequence device_inbox_sequence;
DROP sequence event_auth_chain_id;
DROP sequence events_backfill_stream_seq;
DROP sequence events_stream_seq;
DROP sequence presence_stream_sequence;
DROP sequence receipts_sequence;
DROP sequence un_partial_stated_event_stream_sequence;
DROP sequence un_partial_stated_room_stream_sequence;
```

`\q` to quit

on host:

```
gzip -d synapse_export.sql.gz  
sudo cp synapse_export.sql /data/postgres/synapse/
```

on pod: `cd /var/lib/postgresql/data`

```
pg_restore <connection> --no-owner --role=<new role> -d <new db name> dump.sql
```

Revision #18

Created 19 April 2023 21:04:24 by James Reilly

Updated 27 November 2024 12:05:03 by Kieran Mitchell Lane