1. Installation Seafile 13

We will install Seafile again in our docker vm by using Dockhand. For that we need to make sure that we have a place for all the data, which in our case will be an NFS share hosted on our TrueNas VM. Then the next step is of course to look up the docker compose file and deploy it to our needs.

The docker compose file can be obtained by downloading it to your machine via Seafile 13. People are working on seafile 14 as we speak. So this documentation might be outdated again on the moment you are reading this. Just make sure you check the general website for the latest edition. So this one right now is focused on **Seafile 13**. Once 14 will be available we will update the instructions below.

Now! at this moment I don't have a running seafile 13 yet! So I still need to work on this as a project in the future. What I do realize is that you need to have root running for the storage folder. You can also do this without root but also I don't know yet how to do that. For now make sure your TrueNas NFS share has root as the owner.

1. **Make sure you have compose.yml**
2. **Also you need to run seadoc if you want to have the wiki function. Make sure you copy paste the values from the seafile mysql database**
3. **Create an .env file that serves compose.yml and the seadoc file**

compose.yml

```
services:
  db:
    image: ${SEAFILE_DB_IMAGE:-mariadb:10.11}
    container_name: seafile-mysql
    restart: unless-stopped
    environment:
      - MYSQL_ROOT_PASSWORD=${INIT_SEAFILE_MYSQL_ROOT_PASSWORD:-}
      - MYSQL_LOG_CONSOLE=true
      - MARIADB_AUTO_UPGRADE=1
    volumes:
      - "${SEAFILE_MYSQL_VOLUME:-/opt/seafile-mysql/db}:/var/lib/mysql"
    networks:
      - seafile-net
```

```
  healthcheck:
    test:
      [
        "CMD",
        "/usr/local/bin/healthcheck.sh",
        "--connect",
        "--mariadbupgrade",
        "--innodb_initialized",
      ]
    interval: 20s
    start_period: 30s
    timeout: 5s
    retries: 10

redis:
  image: ${SEAFILE_REDIS_IMAGE:-redis}
  container_name: seafile-redis
  restart: unless-stopped
  command:
    - /bin/sh
    - -c
    - redis-server --requirepass "$$REDIS_PASSWORD"
  environment:
    - REDIS_PASSWORD=${REDIS_PASSWORD:-}
  networks:
    - seafile-net

seafile:
  image: ${SEAFILE_IMAGE:-seafileltd/seafile-mc:13.0-latest}
  container_name: seafile
  restart: unless-stopped
  # ports:
  #   - "80:80"
  volumes:
    - ${SEAFILE_VOLUME:-/opt/seafile-data}:/shared
  environment:
    - SEAFILE_MYSQL_DB_HOST=${SEAFILE_MYSQL_DB_HOST:-db}
    - SEAFILE_MYSQL_DB_PORT=${SEAFILE_MYSQL_DB_PORT:-3306}
    - SEAFILE_MYSQL_DB_USER=${SEAFILE_MYSQL_DB_USER:-seafile}
```

```
      - SEAFILE_MYSQL_DB_PASSWORD=${SEAFILE_MYSQL_DB_PASSWORD:?Variable is not set or
empty}
      - INIT_SEAFILE_MYSQL_ROOT_PASSWORD=${INIT_SEAFILE_MYSQL_ROOT_PASSWORD:-}
      - SEAFILE_MYSQL_DB_CCNET_DB_NAME=${SEAFILE_MYSQL_DB_CCNET_DB_NAME:-ccnet_db}
      - SEAFILE_MYSQL_DB_SEAFILE_DB_NAME=${SEAFILE_MYSQL_DB_SEAFILE_DB_NAME:-seafile_db}
      - SEAFILE_MYSQL_DB_SEAHUB_DB_NAME=${SEAFILE_MYSQL_DB_SEAHUB_DB_NAME:-seahub_db}
      - TIME_ZONE=${TIME_ZONE:-Etc/UTC}
      - INIT_SEAFILE_ADMIN_EMAIL=${INIT_SEAFILE_ADMIN_EMAIL:-me@example.com}
      - INIT_SEAFILE_ADMIN_PASSWORD=${INIT_SEAFILE_ADMIN_PASSWORD:-asecret}
      - SEAFILE_SERVER_HOSTNAME=${SEAFILE_SERVER_HOSTNAME:?Variable is not set or empty}
      - SEAFILE_SERVER_PROTOCOL=${SEAFILE_SERVER_PROTOCOL:-http}
      - SITE_ROOT=${SITE_ROOT:-/}
      - NON_ROOT=${NON_ROOT:-false}
      - JWT_PRIVATE_KEY=${JWT_PRIVATE_KEY:?Variable is not set or empty}
      - SEAFILE_LOG_TO_STDOUT=${SEAFILE_LOG_TO_STDOUT:-false}
      - ENABLE_GO_FILESERVER=${ENABLE_GO_FILESERVER:-true}
      - ENABLE_SEADOC=${ENABLE_SEADOC:-true}
      - SEADOC_SERVER_URL=${SEAFILE_SERVER_PROTOCOL:-
http}://${SEAFILE_SERVER_HOSTNAME:?Variable is not set or empty}/sdoc-server
      - CACHE_PROVIDER=${CACHE_PROVIDER:-redis}
      - REDIS_HOST=${REDIS_HOST:-redis}
      - REDIS_PORT=${REDIS_PORT:-6379}
      - REDIS_PASSWORD=${REDIS_PASSWORD:-}
      - MEMCACHED_HOST=${MEMCACHED_HOST:-memcached}
      - MEMCACHED_PORT=${MEMCACHED_PORT:-11211}
      - ENABLE_NOTIFICATION_SERVER=${ENABLE_NOTIFICATION_SERVER:-false}
      - INNER_NOTIFICATION_SERVER_URL=${INNER_NOTIFICATION_SERVER_URL:-
http://notification-server:8083}
      - NOTIFICATION_SERVER_URL=${NOTIFICATION_SERVER_URL:-${SEAFILE_SERVER_PROTOCOL:-
http}://${SEAFILE_SERVER_HOSTNAME:?Variable is not set or empty}/notification}
      - ENABLE_SEAFILE_AI=${ENABLE_SEAFILE_AI:-false}
      - ENABLE_FACE_RECOGNITION=${ENABLE_FACE_RECOGNITION:-false}
      - SEAFILE_AI_SERVER_URL=${SEAFILE_AI_SERVER_URL:-http://seafile-ai:8888}
      - SEAFILE_AI_SECRET_KEY=${JWT_PRIVATE_KEY:?Variable is not set or empty}
      - MD_FILE_COUNT_LIMIT=${MD_FILE_COUNT_LIMIT:-100000}
    labels:
      caddy: ${SEAFILE_SERVER_PROTOCOL:-http}://${SEAFILE_SERVER_HOSTNAME:?Variable is not
set or empty}
      caddy.reverse_proxy: "{{upstreams 80}}"
```

```
    healthcheck:
      test: ["CMD-SHELL", "curl -f http://localhost:80 || exit 1"]
      interval: 30s
      timeout: 10s
      retries: 3
      start_period: 10s
    depends_on:
      db:
        condition: service_healthy
      redis:
        condition: service_started
    networks:
      - seafile-net

networks:
  seafile-net:
    name: seafile-net
```

.env

```
#################################
# Docker compose configurations #
#################################
COMPOSE_FILE='seafile-server.yml,caddy.yml,seadoc.yml'
COMPOSE_PATH_SEPARATOR=','

## Images
SEAFILE_IMAGE=seafileltd/seafile-mc:13.0-latest
SEAFILE_DB_IMAGE=mariadb:10.11
SEAFILE_REDIS_IMAGE=redis
SEAFILE_CADDY_IMAGE=lucaslorentz/caddy-docker-proxy:2.12-alpine
SEADOC_IMAGE=seafileltd/sdoc-server:2.0-latest
NOTIFICATION_SERVER_IMAGE=seafileltd/notification-server:13.0-latest
```

```
MD_IMAGE=seafileltd/seafile-md-server:13.0-latest

## Persistent Storage
BASIC_STORAGE_PATH=/opt
SEAFILE_VOLUME=$BASIC_STORAGE_PATH/seafile-data
SEAFILE_MYSQL_VOLUME=$BASIC_STORAGE_PATH/seafile-mysql/db
SEAFILE_CADDY_VOLUME=$BASIC_STORAGE_PATH/seafile-caddy
SEADOC_VOLUME=$BASIC_STORAGE_PATH/seadoc-data

#################################
#      Startup parameters       #
#################################
SEAFILE_SERVER_HOSTNAME=seafile.example.com
SEAFILE_SERVER_PROTOCOL=http
TIME_ZONE=Etc/UTC
JWT_PRIVATE_KEY=

#######################################
# Third-party service configuration #
#######################################

## Database
SEAFILE_MYSQL_DB_HOST=db
SEAFILE_MYSQL_DB_USER=seafile
SEAFILE_MYSQL_DB_PASSWORD=PASSWORD
SEAFILE_MYSQL_DB_CCNET_DB_NAME=ccnet_db
SEAFILE_MYSQL_DB_SEAFILE_DB_NAME=seafile_db
SEAFILE_MYSQL_DB_SEAHUB_DB_NAME=seahub_db

## Cache
CACHE_PROVIDER=redis # or memcached


### Redis
REDIS_HOST=redis
REDIS_PORT=6379
REDIS_PASSWORD=

### Memcached
```

```
MEMCACHED_HOST=memcached
MEMCACHED_PORT=11211

#####################################
#        Initial variables          #
# (Only valid in first-time startup) #
#####################################

## Database root password, Used to create Seafile users
INIT_SEAFILE_MYSQL_ROOT_PASSWORD=ROOT_PASSWORD

## Seafile admin user
INIT_SEAFILE_ADMIN_EMAIL=me@example.com
INIT_SEAFILE_ADMIN_PASSWORD=asecret

############################################
# Additional configurations for extensions #
############################## Seafile AI
ENABLE_SEAFILE_AI=false
ENABLE_FACE_RECOGNITION=false
SEAFILE_AI_LLM_TYPE=openai
SEAFILE_AI_LLM_URL=
SEAFILE_AI_LLM_KEY= # your llm key
SEAFILE_AI_LLM_MODEL=gpt-4o-mini

## Metadata server
MD_FILE_COUNT_LIMIT=100000
################

## SeaDoc service
ENABLE_SEADOC=true

## Notification
ENABLE_NOTIFICATION_SERVER=false
NOTIFICATION_SERVER_URL=
```

compose.yml

```
services:
  db:
    image: mariadb:10.11
    container_name: seafile-db
    environment:
      - MYSQL_ROOT_PASSWORD=${SEAFILE_PASSWORD}
      - MYSQL_LOG_CONSOLE=true
      - MARIADB_AUTO_UPGRADE=1
    volumes:
      - ./mysql:/var/lib/mysql
    restart: unless-stopped
    healthcheck:
      test: ["CMD", "healthcheck.sh", "--connect", "--innodb_initialized"]
      interval: 20s
      start_period: 30s
      timeout: 5s
      retries: 10

  redis:
    image: redis:7
    container_name: seafile-redis
    restart: unless-stopped
    healthcheck:
      test: ["CMD", "redis-cli", "ping"]
      interval: 20s
      timeout: 5s
      retries: 5

  seafile:
    image: seafileltd/seafile-mc:13.0-latest
    container_name: seafile
    ports:
      - "1026:80"
    volumes:
      - /mnt/seafile:/shared
    environment:
      # Core
```

```
      - SEAFILE_SERVER_HOSTNAME=${SEAFILE_SERVER_HOSTNAME}
      - SEAFILE_SERVER_PROTOCOL=${SEAFILE_SERVER_PROTOCOL}
      - TIME_ZONE=${TIME_ZONE}
      - JWT_PRIVATE_KEY=${JWT_PRIVATE_KEY}

      # Database
      - DB_HOST=db
      - DB_ROOT_PASSWD=${SEAFILE_PASSWORD}
      - SEAFILE_MYSQL_DB_HOST=db
      - SEAFILE_MYSQL_DB_USER=seafile
      - SEAFILE_MYSQL_DB_PASSWORD=${SEAFILE_PASSWORD}
      - SEAFILE_MYSQL_DB_CCNET_DB_NAME=ccnet_db
      - SEAFILE_MYSQL_DB_SEAFILE_DB_NAME=seafile_db
      - SEAFILE_MYSQL_DB_SEAHUB_DB_NAME=seahub_db

      # Cache
      - CACHE_PROVIDER=redis
      - REDIS_HOST=redis
      - REDIS_PORT=6379

      # First-run admin bootstrap (ignored on subsequent starts)
      - INIT_SEAFILE_ADMIN_EMAIL=${SEAFILE_ADMIN_EMAIL}
      - INIT_SEAFILE_ADMIN_PASSWORD=${SEAFILE_PASSWORD}
      - INIT_SEAFILE_MYSQL_ROOT_PASSWORD=${SEAFILE_PASSWORD}
    depends_on:
      db:
        condition: service_healthy
      redis:
        condition: service_healthy
    restart: unless-stopped
```

.env

```
# ==== Hostname ====
# Bake-in on first run - the domain or IP:port you will actually use to reach Seafile.
# Do NOT change after first boot without also editing conf/seahub_settings.py.
```

```
SEAFILE_SERVER_HOSTNAME=10.69.1.106
SEAFILE_SERVER_PROTOCOL=http
TIME_ZONE=Europe/Amsterdam

# ==== Secrets ====
# Paste the output of: openssl rand -base64 40
JWT_PRIVATE_KEY=m41h+mivVsScLtvik+ebyiPvyjlhiZ1HgeIAjkp5J92aQqnda7uLkQ==

# ONE password used for everything: MariaDB root, Seafile's DB user, and the Seafile web admin.
# Pick a strong unique value.
SEAFILE_PASSWORD=NKetpcGDPqCXGlBYATTTZ26E438Ns1FOPoqlUX0QeOGmgsEYSBToAg==

# Admin email for the first-boot web UI login (only used on first startup).
SEAFILE_ADMIN_EMAIL=dp@p2pnode.org
```

Revision #12
Created 2026-07-18 19:49:48 UTC by dpasn
Updated 2026-07-18 23:10:48 UTC by dpasn