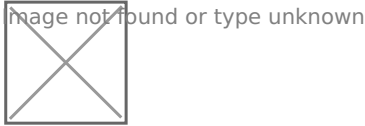


Setup Vikunja using Docker Compose



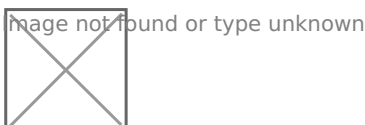
Vikunja is an Open-Source, self-hosted To-Do list application for all platforms. It is licensed under the GPLv3.

Test out [Vikunja demo here](#).

This guide is assuming you are installing Vikunja on a fresh Debian based OS such as Ubuntu, Debian or Turnkey Core. You can edit the docker-compose.yml to change the port and database passwords however, it is not required unless port 80 is being used by another service. **Do NOT modify or change anything in the nginx.conf file** even if you change the port in the docker-compose.yml file.

1. Install docker - apt install docker.io -y
2. Install docker compose - apt install docker-compose -y
3. cd /
4. mkdir docker
5. cd docker
6. mkdir vikunja
7. cd /docker/vikunja
8. create docker-compose.yml - touch docker-compose.yml
9. nano docker-compose.yml - copy the content below into the file then press ctrl x then y then enter.
10. create nginx.conf - touch nginx.conf - If that creates a directory, remove it rm -d nginx.conf then use cat > nginx.conf
11. nano nginx.conf - copy the content from the below example into the file then press ctrl x then y then enter on your keyboard.
12. run docker-compose up -d

When the install finishes you should see these files and folders in /docker/vikunja directory.



Navigate to the IP of your machine and register an account on Vikunja. There is no admin account.

Find the ip by typing the following into the terminal:

```
ip addr
```

Vikunja Docker Compose without Email Notifications

```
version: '3'

services:
  db:
    image: mariadb:10
    command: --character-set-server=utf8mb4 --collation-server=utf8mb4_unicode_ci
    environment:
      MYSQL_ROOT_PASSWORD: supersecret
      MYSQL_USER: vikunja
      MYSQL_PASSWORD: secret
      MYSQL_DATABASE: vikunja
    volumes:
      - ./db: /var/lib/mysql
    restart: unless-stopped
  api:
    image: vikunja/api
    environment:
      VIKUNJA_DATABASE_HOST: db
      VIKUNJA_DATABASE_PASSWORD: secret
      VIKUNJA_DATABASE_TYPE: mysql
      VIKUNJA_DATABASE_USER: vikunja
      VIKUNJA_DATABASE_DATABASE: vikunja
    volumes:
      - ./files: /app/vikunja/files
    depends_on:
      - db
    restart: unless-stopped
  frontend:
    image: vikunja/frontend
    restart: unless-stopped
  proxy:
    image: nginx
```

```
ports:
  - 8022: 80
volumes:
  - ./nginx.conf:/etc/nginx/conf.d/default.conf:ro
depends_on:
  - api
  - frontend
restart: unless-stopped
```

Vikunja Nginx.conf file

```
server {
    listen 80;

    location / {
        proxy_pass http://frontend:80;
    }

    location ~* ^/(api|dav|\.well-known)/ {
        proxy_pass http://api:3456;
        client_max_body_size 20M;
    }
}
```

Vikunja Docker Compose With Email Notifications

The following docker compose is using Google SMTP servers to send email notifications. Edit the SMTP settings in the stack below to suit your needs.

```
version: '3'

services:
  db:
    image: mariadb:10
    command: --character-set-server=utf8mb4 --collation-server=utf8mb4_unicode_ci
    environment:
      MYSQL_ROOT_PASSWORD: supersecret
      MYSQL_USER: vikunja
      MYSQL_PASSWORD: secret
      MYSQL_DATABASE: vikunja
```

```
volumes:
  - ./db: /var/lib/mysql
restart: unless-stopped
api:
  image: vikunja/api
  environment:
    VIKUNJA_DATABASE_HOST: db
    VIKUNJA_DATABASE_PASSWORD: secret
    VIKUNJA_DATABASE_TYPE: mysql
    VIKUNJA_DATABASE_USER: vikunja
    VIKUNJA_DATABASE_DATABASE: vikunja
    VIKUNJA_SERVICE_FRONTENDURL: https://your-frontend-url.com/
    VIKUNJA_SERVICE_ENABLETASKATTACHMENTS: 1
    VIKUNJA_SERVICE_ENABLEREGISTRATION: 0
    VIKUNJA_SERVICE_ENABLEEMAILREMINDERS: 1
    VIKUNJA_MAILER_ENABLED: 1
    VIKUNJA_MAILER_FORCESSL: 1
    VIKUNJA_MAILER_HOST: smtp.gmail.com
    VIKUNJA_MAILER_PORT: 465
    VIKUNJA_MAILER_USERNAME: youremail@gmail.com
    VIKUNJA_MAILER_PASSWORD: yourgmailpassword
  volumes:
    - ./files: /app/vikunja/files
  depends_on:
    - db
  restart: unless-stopped
frontend:
  image: vikunja/frontend
  restart: unless-stopped
proxy:
  image: nginx
  ports:
    - 8022: 80
  volumes:
    - ./nginx.conf: /etc/nginx/conf.d/default.conf:ro
  depends_on:
    - api
    - frontend
  restart: unless-stopped
```

Video Tutorials

https://www.youtube.com/embed/fGlz2PkXjuo?ab_channel=Geeked

Revision #17

Created 22 August 2021 10:05:05 by Jeremy

Updated 4 October 2021 14:18:45 by Jeremy