Komodo - Setup with Postgres

Sebelum menginstal Komo.do, pastikan Docker sudah terinstal. Panduan instalasi Docker dapat ditemukan di tautan berikut : Docker Install CE & Docker Compose Install.

1. Salin `komodo/postgres.compose.yaml` dan `komodo/compose.env` ke host Anda:

```
wget -P komodo https://raw.githubusercontent.com/moghtech/komodo/main/compose/postgres.compose.yaml && \
  wget -P komodo https://raw.githubusercontent.com/moghtech/komodo/main/compose/compose.env
```

2. Edit variabel dalam `komodo/compose.env`.
3. Menyebarkan:

```
docker compose -p komodo -f komodo/postgres.compose.yaml --env-file komodo/compose.env up -d
```

Config Default postgres.compose.yml (Github)

```
###################################
# □□ KOMODO COMPOSE - POSTGRES □□ #
###################################

## This compose file will deploy:
##   1. Postgres + FerretDB Mongo adapter (https://www.ferretdb.com)
##   2. Komodo Core
##   3. Komodo Periphery

services:
  postgres:
    image: postgres:17
    labels:
      komodo.skip: # Prevent Komodo from stopping with StopAllContainers
    restart: unless-stopped
    logging:
      driver: ${COMPOSE_LOGGING_DRIVER:-local}
```

```
  networks:
    - default
  # ports:
  #   - 5432:5432
  volumes:
    - pg-data:/var/lib/postgresql/data
  environment:
    - POSTGRES_USER=${KOMODO_DB_USERNAME}
    - POSTGRES_PASSWORD=${KOMODO_DB_PASSWORD}
    - POSTGRES_DB=${KOMODO_DATABASE_DB_NAME:-komodo}

ferretdb:
  image: ghcr.io/ferretdb/ferretdb:1
  labels:
    komodo.skip: # Prevent Komodo from stopping with StopAllContainers
  restart: unless-stopped
  depends_on:
    - postgres
  logging:
    driver: ${COMPOSE_LOGGING_DRIVER:-local}
  networks:
    - default
  # ports:
  #   - 27017:27017
  environment:
    - FERRETDB_POSTGRESQL_URL=postgres://postgres:5432/${KOMODO_DATABASE_DB_NAME:-komodo}

core:
  image: ghcr.io/mbecker20/komodo:${COMPOSE_KOMODO_IMAGE_TAG:-latest}
  labels:
    komodo.skip: # Prevent Komodo from stopping with StopAllContainers
  restart: unless-stopped
  depends_on:
    - ferretdb
  logging:
    driver: ${COMPOSE_LOGGING_DRIVER:-local}
  networks:
    - default
  ports:
    - 9120:9120
```

```
    env_file: ./compose.env
    environment:
      KOMODO_DATABASE_URI:
mongodb://${KOMODO_DB_USERNAME}:${KOMODO_DB_PASSWORD}@ferretdb:27017/${KOMODO_DATABASE_
DB_NAME:-komodo}?authMechanism=PLAIN
    volumes:
      ## Core cache for repos for latest commit hash / contents
      - repo-cache:/repo-cache
      ## Store sync files on server
      # - /path/to/syncs:/syncs
      ## Optionally mount a custom core.config.toml
      # - /path/to/core.config.toml:/config/config.toml
    ## Allows for systemd Periphery connection at
    ## "http://host.docker.internal:8120"
    # extra_hosts:
    #   - host.docker.internal:host-gateway

  ## Deploy Periphery container using this block,
  ## or deploy the Periphery binary with systemd using
  ## https://github.com/mbecker20/komodo/tree/main/scripts
  periphery:
    image: ghcr.io/mbecker20/periphery:${COMPOSE_KOMODO_IMAGE_TAG:-latest}
    labels:
      komodo.skip: # Prevent Komodo from stopping with StopAllContainers
    restart: unless-stopped
    logging:
      driver: ${COMPOSE_LOGGING_DRIVER:-local}
    networks:
      - default
    env_file: ./compose.env
    environment:
      PERIPHERY_REPO_DIR: ${PERIPHERY_ROOT_DIRECTORY:-/etc/komodo}/repos
      PERIPHERY_STACK_DIR: ${PERIPHERY_ROOT_DIRECTORY:-/etc/komodo}/stacks
      PERIPHERY_SSL_KEY_FILE: ${PERIPHERY_ROOT_DIRECTORY:-/etc/komodo}/ssl/key.pem
      PERIPHERY_SSL_CERT_FILE: ${PERIPHERY_ROOT_DIRECTORY:-/etc/komodo}/ssl/cert.pem
    volumes:
      ## Mount external docker socket
      - /var/run/docker.sock:/var/run/docker.sock
      ## Allow Periphery to see processes outside of container
      - /proc:/proc
```

```
      ## Specify the Periphery agent root directory.
      ## Must be the same inside and outside the container,
      ## or docker will get confused. See https://github.com/mbecker20/komodo/discussions/180.
      ## Default: /etc/komodo.
      - ${PERIPHERY_ROOT_DIRECTORY:-/etc/komodo}:${PERIPHERY_ROOT_DIRECTORY:-/etc/komodo}

volumes:
  # Postgres
  pg-data:
  # Core
  repo-cache:

networks:
  default: {}
```

Config Defauft compose.env ([Github]())

```
###################################
# □□ KOMODO COMPOSE - VARIABLES □□ #
###################################

## These compose variables can be used with all Komodo deployment options.
## Pass these variables to the compose up command using `--env-file komodo/compose.env`.
## Additionally, they are passed to both Komodo Core and Komodo Periphery with `env_file: ./compose.env`,
## so you can pass any additional environment variables to Core / Periphery directly in this file as well.

## Stick to a specific version, or use `latest`
COMPOSE_KOMODO_IMAGE_TAG=latest

## Note: □□ Podman does NOT support local logging driver □□. See Podman options here:
## `https://docs.podman.io/en/v4.6.1/markdown/podman-run.1.html#log-driver-driver`
COMPOSE_LOGGING_DRIVER=local # Enable log rotation with the local driver.

## DB credentials - Ignored for Sqlite
KOMODO_DB_USERNAME=admin
KOMODO_DB_PASSWORD=admin

## Configure a secure passkey to authenticate between Core / Periphery.
KOMODO_PASSKEY=a_random_passkey
```

```
#=--------------------------=#
#= Komodo Core Environment =#
#=--------------------------=#

## Full variable list + descriptions are available here:
## 🦎 https://github.com/mbecker20/komodo/blob/main/config/core.config.toml 🦎

## Note. Secret variables also support `${VARIABLE}_FILE` syntax to pass docker compose secrets.
## Docs: https://docs.docker.com/compose/how-tos/use-secrets/#examples

## Used for Oauth / Webhook url suggestion / Caddy reverse proxy.
KOMODO_HOST=https://demo.komo.do
## Displayed in the browser tab.
KOMODO_TITLE=Komodo
## Create a server matching this address as the "first server".
## Use `https://host.docker.internal:8120` when using systemd-managed Periphery.
KOMODO_FIRST_SERVER=https://periphery:8120
## Make all buttons just double-click, rather than the full confirmation dialog.
KOMODO_DISABLE_CONFIRM_DIALOG=false

## Rate Komodo polls your servers for
## status / container status / system stats / alerting.
## Options: 1-sec, 5-sec, 15-sec, 1-min, 5-min.
## Default: 15-sec
KOMODO_MONITORING_INTERVAL="15-sec"
## Rate Komodo polls Resources for updates,
## like outdated commit hash.
## Options: 1-min, 5-min, 15-min, 30-min, 1-hr.
## Default: 5-min
KOMODO_RESOURCE_POLL_INTERVAL="5-min"

## Used to auth incoming webhooks. Alt: KOMODO_WEBHOOK_SECRET_FILE
KOMODO_WEBHOOK_SECRET=a_random_secret
## Used to generate jwt. Alt: KOMODO_JWT_SECRET_FILE
KOMODO_JWT_SECRET=a_random_jwt_secret

## Enable login with username + password.
KOMODO_LOCAL_AUTH=true
## Disable new user signups.
KOMODO_DISABLE_USER_REGISTRATION=false
```

```
## All new logins are auto enabled
KOMODO_ENABLE_NEW_USERS=false
## Disable non-admins from creating new resources.
KOMODO_DISABLE_NON_ADMIN_CREATE=false
## Allows all users to have Read level access to all resources.
KOMODO_TRANSPARENT_MODE=false

## Time to live for jwt tokens.
## Options: 1-hr, 12-hr, 1-day, 3-day, 1-wk, 2-wk
KOMODO_JWT_TTL="1-day"

## OIDC Login
KOMODO_OIDC_ENABLED=false
## Must reachable from Komodo Core container
# KOMODO_OIDC_PROVIDER=https://oidc.provider.internal/application/o/komodo
## Change the host to one reachable be reachable by users (optional if it is the same as above).
## DO NOT include the `path` part of the URL.
# KOMODO_OIDC_REDIRECT_HOST=https://oidc.provider.external
## Your client credentials
# KOMODO_OIDC_CLIENT_ID= # Alt: KOMODO_OIDC_CLIENT_ID_FILE
# KOMODO_OIDC_CLIENT_SECRET= # Alt: KOMODO_OIDC_CLIENT_SECRET_FILE
## Make usernames the full email.
# KOMODO_OIDC_USE_FULL_EMAIL=true
## Add additional trusted audiences for token claims verification.
## Supports comma separated list, and passing with _FILE (for compose secrets).
# KOMODO_OIDC_ADDITIONAL_AUDIENCES=abc,123 # Alt: KOMODO_OIDC_ADDITIONAL_AUDIENCES_FILE

## Github Oauth
KOMODO_GITHUB_OAUTH_ENABLED=false
# KOMODO_GITHUB_OAUTH_ID= # Alt: KOMODO_GITHUB_OAUTH_ID_FILE
# KOMODO_GITHUB_OAUTH_SECRET= # Alt: KOMODO_GITHUB_OAUTH_SECRET_FILE

## Google Oauth
KOMODO_GOOGLE_OAUTH_ENABLED=false
# KOMODO_GOOGLE_OAUTH_ID= # Alt: KOMODO_GOOGLE_OAUTH_ID_FILE
# KOMODO_GOOGLE_OAUTH_SECRET= # Alt: KOMODO_GOOGLE_OAUTH_SECRET_FILE

## Aws - Used to launch Builder instances and ServerTemplate instances.
KOMODO_AWS_ACCESS_KEY_ID= # Alt: KOMODO_AWS_ACCESS_KEY_ID_FILE
KOMODO_AWS_SECRET_ACCESS_KEY= # Alt: KOMODO_AWS_SECRET_ACCESS_KEY_FILE
```

```
## Hetzner - Used to launch ServerTemplate instances
## Hetzner Builder not supported due to Hetzner pay-by-the-hour pricing model
KOMODO_HETZNER_TOKEN= # Alt: KOMODO_HETZNER_TOKEN_FILE

#=------------------------------=#
#= Komodo Periphery Environment =#
#=------------------------------=#

## Full variable list + descriptions are available here:
## □□ https://github.com/mbecker20/komodo/blob/main/config/periphery.config.toml □□

## Periphery passkeys must include KOMODO_PASSKEY to authenticate.
PERIPHERY_PASSKEYS=${KOMODO_PASSKEY}

## Specify the root directory used by Periphery agent.
PERIPHERY_ROOT_DIRECTORY=/etc/komodo

## Enable SSL using self signed certificates.
## Connect to Periphery at https://address:8120.
PERIPHERY_SSL_ENABLED=true

## If the disk size is overreporting, can use one of these to
## whitelist / blacklist the disks to filter them, whichever is easier.
## Accepts comma separated list of paths.
## Usually whitelisting just /etc/hostname gives correct size.
PERIPHERY_INCLUDE_DISK_MOUNTS=/etc/hostname
# PERIPHERY_EXCLUDE_DISK_MOUNTS=/snap,/etc/repos
```


Revision #2
Created 19 March 2025 03:23:54 by Kevin
Updated 19 March 2025 03:33:35 by Kevin