

Install Hermod

This article describes how to install the [Nexus Personal Messaging](#) component [Hermod](#).

[Expand/Collapse All](#)

Prerequisites

- Knowledge about docker.
- A host with docker and docker-compose installed and configured.
- A public DNS name which devices can reach.
- Matching certificates for the public address.
- Installed instance of an SQL server, such as PostgreSQL, Microsoft SQL Server or Maria DB (unless included in the installation procedure described in this article).

Step-by-step instruction

1. Create a file structure for the configuration and certificate files.

Example: File structure

```
example/  
├── compose  
│   ├── hermod  
│   │   ├── cacerts  
│   │   │   ├── clientcert.cer  
│   │   │   └── certificates  
│   │   ├── hermod-host-bundle.p12  
│   │   ├── config  
│   │   ├── application.yml -> ./cod-hermod.yml  
│   │   ├── cod-hermod.yml  
│   │   └── README.txt  
│   ├── docker-compose.yml  
│   └── postgresql  
└── data  
    └── testdb
```

2. Put your certificates in *example/compose/hermod/certificates* and *example/compose/hermod/cacerts*.



The filenames must not contain spaces and the extension on certificates in *cacerts* should be *.cer*.

Related information

- [Hermod](#)
- [Install Hermod test app](#)
- [Nexus Personal Messaging](#)

The docker image can be loaded locally to each machine, or to the docker registry, and used remotely from several machines.

1. If you want to load the released docker image locally on the target host:

Example: load docker image

```
> docker load -i hermod-2.x.y.RELEASE.tar
```

2. If you have a docker registry, load the image there.

To edit the docker compose configuration:

1. Edit the configuration file `docker-compose.yml` with the correct values for your environment. Below you can find examples, but the actual values must match the specific deployment scenarios.

File path: *example/compose/hermod/docker-compose.yml*

Example: docker-compose.yml

```
version: '2'

services:

  postgresql:
    restart: unless-stopped
    image: postgres:9.4.12
    ports:
      - "5432:5432"
    environment:
      - POSTGRES_USER=dbuser
      - POSTGRES_PASSWORD=dbpassword
      - POSTGRES_DB=hermod
      - TZ=Europe/Stockholm
    mem_limit: 256m
    volumes:
      - ./postgresql/data:/var/lib/postgresql/data:Z

  hermod:
    restart: unless-stopped
    image: ng-docker01.ad.nexusgroup.com:5000/nexus-personal
    /hermod:2.x.y.RELEASE
    ports:
      - "20400:20400"
    extra_hosts:
      - "<my-hermod-server>:94.140.49.248"
    environment:
      - TZ=Europe/Stockholm
      - JAVA_OPTS=-Xms128m -Xmx256m -XX:MaxMetaspaceSize=256m -
XX:CompressedClassSpaceSize=64m -Xss256k -Xmn8m -XX:
InitialCodeCacheSize=4m -XX:ReservedCodeCacheSize=64m -XX:
MaxDirectMemorySize=64m
    mem_limit: 512m
    volumes:
      - ./config:/home/docker/config:z
      - ./certificates:/home/docker/certificates:z
      - ./cacerts:/cacerts:z
    logging:
      options:
        max-size: 10m
    command: "--spring.cloud.config.failFast=false --
application.hermod.swagger.enabled=true"
```

2. The default value for context path and uri base is `/hermod/rest/` (that is `http://localhost:20400/hermod/rest/...`). You might want to change this for your specific configuration. For example, if you want to change the context path and uri base to empty values, change the start command for Hermod in `docker-compose.yml`.

Example: Hermod start command

```
hermod:
  ...
  command: "--server.contextPath=/ ....."
```

Also set the uribase to empty in the Hermod configuration, see the next step.

To edit the Hermod configuration:

1. Edit the configuration file `config/cod-hermod.yml` with the correct values for your environment. Below you can find examples, but the actual values must match the specific deployment scenarios.

Default file path: `example/compose/hermod/config/cod-hermod.yml`

Example: cod-hermod.yml

```
logging:
  level:
    org.springframework.context.annotation.
AnnotationConfigApplicationContext: ERROR
    org.springframework.boot.SpringApplication: ERROR
    org.springframework.cloud.config.client: ERROR
    com.netflix: INFO
    com.nexusgroup: TRACE
    com.relayrides: INFO
  pattern:
    console: "%d{yyyy-MM-dd}T%d{HH:mm:ss.SSS}Z
${LOG_LEVEL_PATTERN:- %5p} [%t] %-40.40logger{39} [%mdc] : %m%
n${LOG_EXCEPTION_CONVERSION_WORD:%wEx}"

server:
  ssl:
    enabled: true
    key-store: /home/docker/certificates/hermod-host-bundle.p12
    key-store-password: "PASSWORD"
    keyStoreType: PKCS12

spring:
  datasource:
    url: jdbc:postgresql://<my-db-server>:5432/hermod
    username: dbuser
    password: dbpasswd

# If the server is exposed on internet without something
# limiting access to endpoints,
# you should only enable specific endpoints and only allow
# management from localhost
endpoints:
  enabled: false
  refresh:
    enabled: true
  health:
    enabled: true
  info:
    enabled: true
```

```
mappings:
  enabled: false

management:
  address: 127.0.0.1
  security:
    enabled: false

application:
  hermod:
    rest:
      log: false

  # Command callback retries
  callback:
    attempts: 1
    retryDelay: 30

  # Hermod clients/users. Connecting clients must set X-API-
  Key
  allowedClients:
    # Note!
    # The X-API-Key should be created using base64(clientId:
  key)
    #
    # Hermod has a helper endpoint to generate configuration.
  Simply use (make sure you have the correct host/port)
    # curl 'http://localhost:20400/hermod/rest/util
  /generateclient/default'
    # to get a snippet which can be pasted to the
  configuration file
    #
    # X-API-Key:
  ZGVmYXVsdDowZTEyYjNhMTgxYzQ0N2YxYjdkMTc0NTg1OGQ4NTgzZTE5Nzc0M2Ri
  NTY2MzQ0N2E5Y2Q5OWI1ZDc1NDhiMThj
    - clientId: default
    key:
  0e12b3a181c447f1b7d1745858d8583e197743db5663447a9cd99b5d7548b18c
    # Optional username:password to be supplied for basic
  authentication in callbacks
    # callbackBasicAuth: username:password
    # The callback URL base for this specific client
    callbackUrl: http://localhost:20400/hermod/rest
    # X-API-Key:
  aGVybW9kLXRlc3Rhcnh6MjY5NzJkOGZhOTQxNGI4MmJmMzVjYzllNGI3YmY2NWU1
  MWZiYjEzNGFiMjY0MGFlYmJkM2U3N2U3ZjE0NDAwMg==
    - clientId: hermod-testapp
    key:
  26972d8fa9414b81bf35cc9e4b7bf65e51fbb134ab2640aeabd3e77e7f144002
    # Optional username:password to be supplied for basic
  authentication in callbacks
    # callbackBasicAuth: username:password
    # The callback URL base for this specific client
    callbackUrl: https://<my-hermod-server>:20488/hermod-
  testapp/rest

  # Message server library settings
  messageServerLibrary:
```

```
# Public URL to the 'south-bound' message API. Please
note that this URL must be reachable
# from your devices. Try to access it from the browser on
your phone
# It can also be the URL to a proxy fronting Hermod (with
SSL offloading etc)
publicUrl: https://<my-hermod-server>:20400/hermod/rest/ms

anyCreate:
  allowed: false

# A string to use for the {@code X-Powered-By} header.
poweredby: Nexus
```

2. If you want to change the uribase, set the corresponding variable, for example to empty, in the *cod-hermod.yml* configuration file:

Set uribase to empty

```
application:
  hermod:
    rest:
      uribase: ""
```

1. Start Postgres (unless you have your own, external instance running):

Example: Start Postgres

```
> cd example/compose/hermod
> docker-compose up -d postgresql
```

2. Connect to your database and run the SQL ddl for the specific vendor/version (for example, *postgresql/V2.3.5_base_version.sql*).
3. Start Hermod with the following command:

Example: Start hermod and hermod-testapp

```
> docker-compose up -d hermod
> docker-compose logs -f
```

4. To install and start Hermod test app, which is a GUI that can be used for basic tests, see [Install Hermod test app](#).