Local installation instructions

18 November 2020

CONTEXT

HONEUR (Haematology Outcomes Network in EURope), a network of Haematology centers, is being established with the mutual goal of being able to track treatment outcomes in those centres and generate RWD to create valuable RWE. Each member of the network will sign a contract with Janssen agreeing to conduct studies on their RWE data and share the results.

HONEUR is a federated data network of haematology centers to support real world evidence research with a focus on outcomes management. The focus is on sharing analysis methods and results, not the granular data itself. To accommodate this objective, every data source is converted to the same data model (aka a Common Data Model - CDM). This is done locally at the participating centers, and, importantly, the data stays locally.

Since all centers are using the same model, a data analysis script can be defined centrally and sent to each of the participating centers for performing the analysis against their local data. The summary results from the individual centers can then be shared with participants through the central database, and summary analysis and visualizations can be done in the central workbench.

REQUIREMENTS

Hardware

Modern 64 bit AMD dual core processor (or better)
8 GB RAM, 16 GB RAM recommended
100 GB free disk space (or more)

Operating system

Windows 10, MacOS or Linux (Ubuntu, CentOS, Debian, …)

Docker

- Windows: https://docs.docker.com/docker-for-windows/install/

- MacOS: https://docs.docker.com/docker-for-mac/install/

- Linux: https://docs.docker.com/install/linux/docker-ce/ubuntu/

Assign 2 or more CPU's, 8 GB of RAM and 100 GB of disk space to Docker in Docker Desktop.

On Linux Docker compose (v1.24 or higher) should be installed separately.

Docker images for HONEUR

A Docker Hub account with read access on the HONEUR Docker image repository
(https://hub.docker.com/u/honeur) is required.

Please create a Docker hub account or use an existing account and request access by sending a mail to
Michel Van Speybroeck (mvspeybr@its.jnj.com)

INSTALLATION COMPONENTS

The HONEUR local installation consists of the following components.

PostgreSQL database

The Postgres image for HONEUR contains the OHDSI database with 4 predefined schema's:

omopcdm
> OMOP CDM (v5.3.1) schema containing the vocabulary and custom concepts for HONEUR.

webapi
> Schema used by the backend of Atlas (WebAPI) to store application data

results
> Schema used by the backend of Atlas (WebAPI) to store result data

scratch
> Empty schema that can be freely used

See https://www.ohdsi.org/data-standardization/the-common-data-model/ for more info about OMOP CDM

User Management

Optional component that can be used to enable security based on a local user database (as an alternative for
integrating with an LDAP server).

User Management is a simple web application to create users, roles and permissions and assign permissions
to roles and roles to users.

More information can be found on the HONEUR portal:
https://portal.honeur.org/group/honeur/knowledge/-/knowledge_base/user-documentation/user-management
-at-the-remote-site

Altas / WebAPI

Atlas is used to explore data in the OMOP CDM database. It's is also used to import cohort definitions that
are shared, run them on the local data and export the inclusion results back to the HONEUR central site.

More information can be found on the HONEUR portal
(https://portal.honeur.org/group/honeur/knowledge/-/knowledge_base/user-documentation/atlas-and-webapi
-at-the-remote-site) and on the OHDSI website (https://github.com/OHDSI/Atlas/wiki)

Zeppelin

Zeppelin is a notebook server that is used to import and run notebooks that are shared. The results after
running the analytics can be uploaded to the central side (manually or automatically). This component is also
used for preparing analytical data for the distributed analytics system if your organization participates in a
distributed analytics study.

More information can be found on the HONEUR portal:
https://portal.honeur.org/group/honeur/knowledge/-/knowledge_base/user-documentation/zeppelin-at-the-re
mote-site and on the Zeppelin website https://zeppelin.apache.org/

Distributed Analytics Remote

Distributed Analytics Remote is an optional component that is listening for distributed request messages when
your organization is participating in a distributed analytics study. Requests are propagated to the
computational Distributed Analytics R-Server component (see below). Results of this computation are sent
back to the central server where the analyst of the distributed analytics study can examine these results.

Distributed Analytics R-Server

Distributed Analytics R-Server is an optional computational component. This component can analyze the data
that is prepared in context of a distributed request and send results of these computations to the distributed
analytics remote component.

HONEUR Studio

HONEUR Studio is an optional component that provides Visual Studio Code, R Studio Server and Shiny
Server. It is possible to import and run Python and R code from within a browser. When authentication is
enabled during the installation, each (local) user will have a personal web space.

HONEUR Proxy

HONEUR Proxy is an optional component that provides a portal page that makes it easier to access the
different HONEUR components that are installed. HONEUR Proxy is a NGINX server. The proxy makes it
possible to access all HONEUR applications on HTTP port 80 (or HTTPS port 443 if a certificate is available).
As an example, Atlas can be accessed on the host machine @ http://localhost/atlas/ instead of @
http://localhost:8080/atlas/. There's no need to open ports 8080, 8081 and 8082 on the host machine if the
proxy is installed.

INSTALLATION OPTIONS

The HONEUR local components can be installed with security enabled or disabled.

When security is disabled, Atlas and Zeppelin (and other optional components if applicable) can be accessed without login.

When security is required, the installer can choose to integrate with a local LDAP server (if available) or use the security database that comes out of the box with the secure setup.

INSTALLATION EFFORT

Installing the local HONEUR components doesn't take more than 15 minutes once a (physical or virtual) host machine - with Docker installed on it - is available.

INSTALLATION STEPS

Full installation instructions are documented on the official github page
https://github.com/solventrix/Honeur-Setup/tree/master/remote-installation

Installation Notes

Downloading the Docker images can take some time.
The following ports are used on the host machine:

• Postgres: 5444

• HONEUR Proxy: 80

Please contact the HONEUR support team if these ports are not available or should be changed. If the HONEUR Proxy is installed, only HTTP port 80 (and/or HTTPS port 443) should be available on the host machine.
The database data will be stored in a Docker volume named "pgdata"

HOW TO ACCESS THE HONEUR INSTALLATION

After the installation is completed. All HONEUR components are available and can be accessed as follows:

Postgres DB

1. Open a SQL Client (e.g.: https://dbeaver.io/)
2. Connect to the Postgres database running on:
   Host: hostname or IP address of the host machine
   Port: **5444**
   Database: **OHDSI**
   Username: honeur, Password: honeur (to select, update and delete data) or
   Username: honeur_admin, Password: honeur_admin (to make DB changes)
   ***Note:** Please change the default password of the honeur and honeur_admin DB accounts*
   Changing the password of the honeur and honeur_admin account will not have impact on the
   correct working of the installation.
3. The "omopcdm", "results", "webapi" and scratch schemas can be accessed in the database.

HONEUR Proxy

1. Open a modern browser (e.g. Google Chrome)
2. Navigate to **http://[[hostname or IP of host machine]**
3. The HONEUR local portal page will be displayed. Atlas, Zeppelin, User Management (if applicable), Visual Studio Code (if applicable) and R Studio server (if applicable) can be opened via this portal page.

Atlas

1. Open a modern browser (e.g. Google Chrome)
2. Navigate to **http://[hostname or IP of host machine]/atlas**
3. The Atlas home page will be displayed.

User Management (optional)

1. Open a modern browser (e.g. Google Chrome)
2. Navigate to **http://[hostname or IP of host machine]/user-mgmt**
3. The user management home page will be displayed.

Zeppelin

1. Open a modern browser (e.g. Google Chrome)
2. Navigate to **http://[hostname or IP of host machine]/zeppelin**
3. The Zeppelin home page will be displayed.

HONEUR Studio - VSCode

4. Open a modern browser (e.g. Google Chrome)
5. Navigate to **http://[hostname or IP of host machine]**/honeur-studio/app/vscode
6. VSCode IDE will be displayed.

HONEUR Studio - RStudio

7. Open a modern browser (e.g. Google Chrome)
8. Navigate to **http://[hostname or IP of host machine]**/honeur-studio/app/rstudio
9. RStudio IDE will be displayed.

HONEUR Studio - Local Shiny Apps

10. Open a modern browser (e.g. Google Chrome)
11. Navigate to **http://[hostname or IP of host machine]**/honeur-studio/app/reports
12. A list of locally deployed shiny apps will be displayed.

HONEUR Studio - Local Documents

13. Open a modern browser (e.g. Google Chrome)
14. Navigate to **http://[hostname or IP of host machine]**/honeur-studio/app/documents
15. A list of local documents will be displayed.

HONEUR Studio -personal

16. Open a modern browser (e.g. Google Chrome)
17. Navigate to **http://[hostname or IP of host machine]**/honeur-studio/app/personal
18. A list of files in the personal folder will be displayed.

RUNNING THE ETL

The ETL code to map the source data to the OMOP Common Data Model (CDM) will be custom build for each data center. The script to execute the ETL will be provided or you can check https://github.com/solventrix/Honeur-Setup under "Run ETL"

POST ETL INSTALLATION STEPS

Add constraints and indexes to the OMOP CDM tables

After the ETL is successfully executed, it's recommended to add the constraints and indexes to the OMOP CDM tables. It will improve the performance and reduce the risk of corrupt data in the database.

Installation steps

1. Open a terminal window (Command Prompt on Windows)

2. Download the installation file

   a. For Windows:
   ```
   curl -L
   https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/start-omop
   cdm-indexes-and-constraints.cmd --output
   start-omopcdm-indexes-and-constraints.cmd
   ```

   b. For Linux and Mac
   ```
   curl -L
   https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/start-omop
   cdm-indexes-and-constraints.sh --output
   start-omopcdm-indexes-and-constraints.sh && chmod +x
   start-omopcdm-custom-concepts-update.sh
   ```

1. Run `start-omopcdm-indexes-and-constraints.sh` (on Linux or Mac) or
   `start-omopcdm-indexes-and-constraints.cmd` (on Windows)

   > The program will prompt to enter the username and password of your Docker account. Make sure this Docker account has read access on the HONEUR images. If you are already logged in to Docker, the script will automatically use the existing credentials. Press Enter to remove existing omop-indexes-and-constraints container (if present). After the script has downloaded the docker-compose.yml and setup-conf/setup.yml files, it will start creating the indexes and constraints on the OMOP CDM tables.

Update custom concepts in the OMOP CDM DB

When new custom concepts are available, they can be easily loaded in the OMOP CDM database.

Installation steps

1. Open a terminal window (Command Prompt on Windows)
2. Download the installation file
   - For Windows:
   ```
   curl -L
   https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/sta
   rt-omopcdm-custom-concepts-update.cmd --output
   start-omopcdm-custom-concepts-update.cmd
   ```
   - For Linux and Mac:

```
curl -L
https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/sta
rt-omopcdm-custom-concepts-update.sh --output
start-omopcdm-custom-concepts-update.sh && chmod +x
start-omopcdm-custom-concepts-update.sh
```

3. Run start-omopcdm-custom-concepts-update.sh (on Linux or Mac) or start-omopcdm-custom-concepts-update.cmd (on Windows)
4. The program will prompt you for username and password for your docker account. Make sure this docker account has read access on the honeur images. If you are already logged in to docker, the program will automatically use the existing credentials.
5. Press Enter to remove the existing omop-cdm-custom-concepts container.
6. After the program has downloaded the docker-compose.yml and setup-conf/setup.yml files, it will start importing the custom concepts in the OMOP CDM database.

QA database

QA database can be used as a test database. It's an exact replica of the full database installed with the script start-honeur.cmd (on windows) or start-honeur.sh (on Linux or Mac). It is primarily used for testing scripts on the data in OMOP CDM schema.

Installation steps

1. Open a terminal window (Command Prompt on Windows)
2. Download the installation file
   - For Windows:
     ```
     curl -L
     https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/sta
     rt-qa-database.cmd --output start-qa-database.cmd
     ```
   - For Linux and Mac:
     ```
     curl -L
     https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/sta
     rt-qa-database.sh --output start-qa-database.sh && chmod +x
     start-qa-database.sh
     ```
3. Run start-qa-database.sh (on Linux or Mac) or start-qa-database.cmd (on Windows)
4. The program will prompt you for username and password for your docker account. Make sure this docker account has read access on the honeur images. If you are already logged in to docker, the program will automatically use the existing credentials.
5. Press Enter to remove existing postgres-qa and webapi-source-qa-enable container.
6. After the program has downloaded the docker-compose.yml and setup-conf/setup.yml files, it will start initializing the QA database and insert the source of the QA database inside the original database.
7. Restart webapi/atlas container to make the source available in the webapi/atlas instance. Do this using the following command: *docker restart webapi*

Removal steps

1. Open a terminal window (Command Prompt on Windows)
2. Download the installation file
   - For Windows:
     ```
     curl -L
     https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/rem
     ove-qa-database.cmd --output remove-qa-database.cmd
     ```
   - For Linux and Mac:
     ```
     curl -L
     https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/rem
     ove-qa-database.sh --output remove-qa-database.sh && chmod +x
     remove-qa-database.sh
     ```
3. Run remove-qa-database.sh (on Linux or Mac) or remove-qa-database.cmd (on Windows)
4. The program will prompt you for username and password for your docker account. Make sure this docker account has read access on the honeur images. If you are already logged in to docker, the program will automatically use the existing credentials.
5. Press Enter to remove existing postgres-qa and webapi-source-qa-disable container.
6. After the program has downloaded the docker-compose.yml and setup-conf/setup.yml files, it will start removing the QA database and removing the source of the QA database inside the original database.

MAINTENANCE

Starting and stopping the Docker Containers

The Docker containers for HONEUR will automatically restart when the host machine is restarted. To manually start and stop the containers, open a terminal window and run one of the following commands.

Postgres

- **to start**: `docker start postgres`
- **to stop**: `docker stop postgres`

Atlas

- **to start**: `docker start webapi`
- **to stop**: `docker stop webapi`

Zeppelin

- **to start**: `docker start zeppelin`
- **to stop**: `docker stop zeppelin`

User Management

- **to start**: `docker start user-mgmt`
- **to stop**: `docker stop user-mgmt`

HONEUR Proxy

- **to start**: `docker start nginx`
- **to stop**: `docker stop nginx`

HONEUR Studio

- **to start**: `docker start honeur-studio`
- **to stop**: `docker stop honeur-studio`

HONEUR Studio chronicle

- **to start**: `docker start honeur-studio-chronicle`
- **to stop**: `docker stop honeur-studio-chronicle`

Distributed Analytics Remote

- **to start**: `docker start distributed-analytics-remote`
- **to stop**: `docker stop distributed-analytics-remote`

Distributed Analytics R-Server

- **to start**: `docker start distributed-analytics-r-server`
- **to stop**: `docker stop distributed-analytics-r-server`

Backup and restore of the database

Backup

Download and run the script 'backup-database.sh':

```
curl -L
https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/backup-database.
sh --output backup-database.sh && chmod +x backup-database.sh
```

The backup script will create a tar file name 'postgres_backup_<date_time>.tar.bz2' in the current directory. Creating the backup file can take a long time depending on the size of the database.

Copy the backup file to a save location for long term storage.

Restore

Download the script 'restore-database.sh'

```
curl -L
https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/restore-database
.sh --output restore-database.sh && chmod +x restore-database.sh
```

Run the script and provide the name of the backup file as parameter. The name of the target volume can be provided as a second argument. The backup will be restored in the pgdata volume if the target volume is not provided. The backup file should be present in the folder where the script is executed.

```
./restore-database postgres_backup_<date_time>.tar.bz2
```

Hot snapshot of the database

The database volume can be copied to a new volume (with a different name) to take a snapshot of the current database state. Download the script 'clone-volume.sh'

```
curl -L
https://raw.githubusercontent.com/solventrix/Honeur-Setup/v2.0.0/clone-volume.sh
--output clone-volume.sh && chmod +x clone-volume.sh
```

Run the script, provide the source volume as first parameter and the target volume as second parameter.

```
./clone-volume.sh pgdata pgdata_snapshot1
```

Accessing log files

Console logs

Docker provides real time access to the applications logs.

- Atlas: `docker logs webapi -f`
- Zeppelin: `docker logs zeppelin -f`
- Posgres: `docker logs postgres -f`
- HONEUR Proxy: `docker logs nginx -f`
- HONEUR Studio: `docker logs honeur-studio -f`

Logs files

The log files of Zeppelin are available on the host machine where the Docker containers are running. The location of the logs files can be configured during the setup. The default location is /zeppelin/logs

The logs files of Atlas are available within the Docker container and can be copied to the host machine if needed. The logs files are available in the /usr/local/tomcat/logs directory.

Execute the following commands to inspect the log files:

- `docker exec -it webapi /bin/bash`
- `cd logs`
- `tail -200  webapi.log` or `tail catalina.<yyyy-mm-dd>.log`

Execute the following command to copy all log files to a logs directory below the current directory on host machine:

- `docker cp webapi:/usr/local/tomcat/logs .`

FURTHER READING

More documentation is available in the user documentation on the HONEUR portal:
https://portal.honeur.org/group/honeur/knowledge

ADDENDUM:
SECURITY SCAN OF DOCKER CONTAINERS

The following images were scanned for vulnerabilities:
- honeur/postgres:HONEUR-9.6-omopcdm-5.3.1-webapi-2.7.1-2.0.0
- honeur/webapi-atlas:2.7.1-2.0.0-standard
- honeur/user-mgmt:2.0.0
- honeur/zeppelin:0.8.2-2.0.0
- honeur/distributed-analytics:remote-2.0.0
- honeur/distributed-analytics:r-server-2.0.0
- honeur/honeur-studio:2.0.0
- honeur/nginx:2.0.0

Vulnerability scanner

Docker bench security is used for the vulnerability scan on the images and containers.
https://github.com/docker/docker-bench-security
The Docker Bench for Security is a script that checks for dozens of common best-practices around deploying Docker containers in production. The tests are all automated and are inspired by the CIS Docker Community Edition Benchmark v1.2.0.

Results

honeur/postgres:HONEUR-9.6-omopcdm-5.3.1-webapi-2.7.1-2.0.0

Command

```
git clone https://github.com/docker/docker-bench-security.git

cd docker-bench-security

sudo sh docker-bench-security.sh -i

honeur/postgres:HONEUR-9.6-omopcdm-5.3.1-webapi-2.7.1-2.0.0,postgres -c

container_images,container_runtime
```

Output

[INFO] 4 - Container Images and Build File
[PASS] 4.1 - Ensure that a user for the container has been created (Scored)
[NOTE] 4.2 - Ensure that containers use only trusted base images (Not Scored)
[NOTE] 4.3 - Ensure that unnecessary packages are not installed in the container (Not Scored)
[NOTE] 4.4 - Ensure images are scanned and rebuilt to include security patches (Not Scored)
[WARN] 4.5 - Ensure Content trust for Docker is Enabled (Scored)
[PASS] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images (Scored)
[INFO] 4.7 - Ensure update instructions are not use alone in the Dockerfile (Not Scored)
[INFO]   * Update instruction found: [honeur/postgres:HONEUR-9.6-omopcdm-5.3.1-webapi-2.7.1-2.0.0]
[NOTE] 4.8 - Ensure setuid and setgid permissions are removed (Not Scored)

[PASS] 4.9 - Ensure that COPY is used instead of ADD in Dockerfiles (Not Scored)
[NOTE] 4.10 - Ensure secrets are not stored in Dockerfiles (Not Scored)
[NOTE] 4.11 - Ensure only verified packages are are installed (Not Scored)

[INFO] 5 - Container Runtime
[WARN] 5.1 - Ensure that, if applicable, an AppArmor Profile is enabled (Scored)
[WARN]   * No AppArmorProfile Found: postgres
[PASS] 5.2 - Ensure that, if applicable, SELinux security options are set (Scored)
[PASS] 5.3 - Ensure that Linux kernel capabilities are restricted within containers (Scored)
[PASS] 5.4 - Ensure that privileged containers are not used (Scored)
[PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers (Scored)
[PASS] 5.6 - Ensure sshd is not run within containers (Scored)
[PASS] 5.7 - Ensure privileged ports are not mapped within containers (Scored)
[NOTE] 5.8 - Ensure that only needed ports are open on the container (Not Scored)
[PASS] 5.9 - Ensure that the host's network namespace is not shared (Scored)
[PASS] 5.10 - Ensure that the memory usage for containers is limited (Scored)
[PASS] 5.11 - Ensure that CPU priority is set appropriately on containers (Scored)
[WARN] 5.12 - Ensure that the container's root filesystem is mounted as read only (Scored)
[WARN]   * Container running with root FS mounted R/W: postgres
[WARN] 5.13 - Ensure that incoming container traffic is bound to a specific host interface (Scored)
[WARN]   * Port being bound to wildcard IP: 0.0.0.0 in postgres
[PASS] 5.14 - Ensure that the 'on-failure' container restart policy is set to '5' (Scored)
[PASS] 5.15 - Ensure that the host's process namespace is not shared (Scored)
[PASS] 5.16 - Ensure that the host's IPC namespace is not shared (Scored)
[PASS] 5.17 - Ensure that host devices are not directly exposed to containers (Not Scored)
[PASS] 5.18 - Ensure that the default ulimit is overwritten at runtime if needed (Not Scored)
[PASS] 5.19 - Ensure mount propagation mode is not set to shared (Scored)
[PASS] 5.20 - Ensure that the host's UTS namespace is not shared (Scored)
[PASS] 5.21 - Ensure the default seccomp profile is not Disabled (Scored)
[NOTE] 5.22 - Ensure that docker exec commands are not used with the privileged option (Scored)
[NOTE] 5.23 - Ensure that docker exec commands are not used with the user=root option (Not Scored)
[PASS] 5.24 - Ensure that cgroup usage is confirmed (Scored)
[PASS] 5.25 - Ensure that the container is restricted from acquiring additional privileges (Scored)
[PASS] 5.26 - Ensure that container health is checked at runtime (Scored)
[INFO] 5.27 - Ensure that Docker commands always make use of the latest version of their image (Not Scored)
[PASS] 5.28 - Ensure that the PIDs cgroup limit is used (Scored)
[INFO] 5.29 - Ensure that Docker's default bridge docker0 is not used (Not Scored)
[INFO]   * Container in docker0 network: postgres
[PASS] 5.30 - Ensure that the host's user namespaces are not shared (Scored)
[PASS] 5.31 - Ensure that the Docker socket is not mounted inside any containers (Scored)

honeur/webapi-atlas:2.7.1-2.0.0-standard

Command

```
git clone https://github.com/docker/docker-bench-security.git

cd docker-bench-security

sudo sh docker-bench-security.sh -i

honeur/webapi-atlas:2.7.1-2.0.0-standard,webapi -c

container_images,container_runtime
```

Output

[INFO] 4 - Container Images and Build File
[PASS] 4.1 - Ensure that a user for the container has been created (Scored)

[NOTE] 4.2 - Ensure that containers use only trusted base images (Not Scored)
[NOTE] 4.3 - Ensure that unnecessary packages are not installed in the container (Not Scored)
[NOTE] 4.4 - Ensure images are scanned and rebuilt to include security patches (Not Scored)
[WARN] 4.5 - Ensure Content trust for Docker is Enabled (Scored)
[PASS] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images (Scored)
[INFO] 4.7 - Ensure update instructions are not use alone in the Dockerfile (Not Scored)
[INFO]   * Update instruction found: [honeur/webapi-atlas:2.7.1-2.0.0-standard]
[NOTE] 4.8 - Ensure setuid and setgid permissions are removed (Not Scored)
[PASS] 4.9 - Ensure that COPY is used instead of ADD in Dockerfiles (Not Scored)
[NOTE] 4.10 - Ensure secrets are not stored in Dockerfiles (Not Scored)
[NOTE] 4.11 - Ensure only verified packages are are installed (Not Scored)

[INFO] 5 - Container Runtime
[WARN] 5.1 - Ensure that, if applicable, an AppArmor Profile is enabled (Scored)
[WARN]   * No AppArmorProfile Found: webapi
[PASS] 5.2 - Ensure that, if applicable, SELinux security options are set (Scored)
[PASS] 5.3 - Ensure that Linux kernel capabilities are restricted within containers (Scored)
[PASS] 5.4 - Ensure that privileged containers are not used (Scored)
[PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers (Scored)
[PASS] 5.6 - Ensure sshd is not run within containers (Scored)
[PASS] 5.7 - Ensure privileged ports are not mapped within containers (Scored)
[NOTE] 5.8 - Ensure that only needed ports are open on the container (Not Scored)
[PASS] 5.9 - Ensure that the host's network namespace is not shared (Scored)
[PASS] 5.10 - Ensure that the memory usage for containers is limited (Scored)
[PASS] 5.11 - Ensure that CPU priority is set appropriately on containers (Scored)
[WARN] 5.12 - Ensure that the container's root filesystem is mounted as read only (Scored)
[WARN]   * Container running with root FS mounted R/W: webapi
[PASS] 5.13 - Ensure that incoming container traffic is bound to a specific host interface (Scored)
[PASS] 5.14 - Ensure that the 'on-failure' container restart policy is set to '5' (Scored)
[PASS] 5.15 - Ensure that the host's process namespace is not shared (Scored)
[PASS] 5.16 - Ensure that the host's IPC namespace is not shared (Scored)
[PASS] 5.17 - Ensure that host devices are not directly exposed to containers (Not Scored)
[PASS] 5.18 - Ensure that the default ulimit is overwritten at runtime if needed (Not Scored)
[PASS] 5.19 - Ensure mount propagation mode is not set to shared (Scored)
[PASS] 5.20 - Ensure that the host's UTS namespace is not shared (Scored)
[PASS] 5.21 - Ensure the default seccomp profile is not Disabled (Scored)
[NOTE] 5.22 - Ensure that docker exec commands are not used with the privileged option (Scored)
[NOTE] 5.23 - Ensure that docker exec commands are not used with the user=root option (Not Scored)
[PASS] 5.24 - Ensure that cgroup usage is confirmed (Scored)
[PASS] 5.25 - Ensure that the container is restricted from acquiring additional privileges (Scored)
[PASS] 5.26 - Ensure that container health is checked at runtime (Scored)
[INFO] 5.27 - Ensure that Docker commands always make use of the latest version of their image (Not Scored)
[PASS] 5.28 - Ensure that the PIDs cgroup limit is used (Scored)
[INFO] 5.29 - Ensure that Docker's default bridge docker0 is not used (Not Scored)
[INFO]   * Container in docker0 network: webapi
[PASS] 5.30 - Ensure that the host's user namespaces are not shared (Scored)
[PASS] 5.31 - Ensure that the Docker socket is not mounted inside any containers (Scored)

honeur/webapi-atlas:2.7.1-2.0.0-secure

Command

```
git clone https://github.com/docker/docker-bench-security.git

cd docker-bench-security

sudo sh docker-bench-security.sh -i honeur/webapi-atlas:2.7.1-2.0.0-secure,webapi

-c container_images,container_runtime
```

Output

[INFO] 4 - Container Images and Build File
[PASS] 4.1 - Ensure that a user for the container has been created (Scored)
[NOTE] 4.2 - Ensure that containers use only trusted base images (Not Scored)
[NOTE] 4.3 - Ensure that unnecessary packages are not installed in the container (Not Scored)
[NOTE] 4.4 - Ensure images are scanned and rebuilt to include security patches (Not Scored)
[WARN] 4.5 - Ensure Content trust for Docker is Enabled (Scored)
[PASS] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images (Scored)
[INFO] 4.7 - Ensure update instructions are not use alone in the Dockerfile (Not Scored)
[INFO]   * Update instruction found: [honeur/webapi-atlas:2.7.1-2.0.0-secure]
[NOTE] 4.8 - Ensure setuid and setgid permissions are removed (Not Scored)
[PASS] 4.9 - Ensure that COPY is used instead of ADD in Dockerfiles (Not Scored)
[NOTE] 4.10 - Ensure secrets are not stored in Dockerfiles (Not Scored)
[NOTE] 4.11 - Ensure only verified packages are are installed (Not Scored)

[INFO] 5 - Container Runtime
[WARN] 5.1 - Ensure that, if applicable, an AppArmor Profile is enabled (Scored)
[WARN]   * No AppArmorProfile Found: webapi
[PASS] 5.2 - Ensure that, if applicable, SELinux security options are set (Scored)
[PASS] 5.3 - Ensure that Linux kernel capabilities are restricted within containers (Scored)
[PASS] 5.4 - Ensure that privileged containers are not used (Scored)
[PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers (Scored)
[PASS] 5.6 - Ensure sshd is not run within containers (Scored)
[PASS] 5.7 - Ensure privileged ports are not mapped within containers (Scored)
[NOTE] 5.8 - Ensure that only needed ports are open on the container (Not Scored)
[PASS] 5.9 - Ensure that the host's network namespace is not shared (Scored)
[PASS] 5.10 - Ensure that the memory usage for containers is limited (Scored)
[PASS] 5.11 - Ensure that CPU priority is set appropriately on containers (Scored)
[WARN] 5.12 - Ensure that the container's root filesystem is mounted as read only (Scored)
[WARN]   * Container running with root FS mounted R/W: webapi
[PASS] 5.13 - Ensure that incoming container traffic is bound to a specific host interface (Scored)
[PASS] 5.14 - Ensure that the 'on-failure' container restart policy is set to '5' (Scored)
[PASS] 5.15 - Ensure that the host's process namespace is not shared (Scored)
[PASS] 5.16 - Ensure that the host's IPC namespace is not shared (Scored)
[PASS] 5.17 - Ensure that host devices are not directly exposed to containers (Not Scored)
[PASS] 5.18 - Ensure that the default ulimit is overwritten at runtime if needed (Not Scored)
[PASS] 5.19 - Ensure mount propagation mode is not set to shared (Scored)
[PASS] 5.20 - Ensure that the host's UTS namespace is not shared (Scored)
[PASS] 5.21 - Ensure the default seccomp profile is not Disabled (Scored)
[NOTE] 5.22 - Ensure that docker exec commands are not used with the privileged option (Scored)
[NOTE] 5.23 - Ensure that docker exec commands are not used with the user=root option (Not Scored)
[PASS] 5.24 - Ensure that cgroup usage is confirmed (Scored)
[PASS] 5.25 - Ensure that the container is restricted from acquiring additional privileges (Scored)
[PASS] 5.26 - Ensure that container health is checked at runtime (Scored)
[INFO] 5.27 - Ensure that Docker commands always make use of the latest version of their image (Not Scored)
[PASS] 5.28 - Ensure that the PIDs cgroup limit is used (Scored)
[INFO] 5.29 - Ensure that Docker's default bridge docker0 is not used (Not Scored)
[INFO]   * Container in docker0 network: webapi
[PASS] 5.30 - Ensure that the host's user namespaces are not shared (Scored)
[PASS] 5.31 - Ensure that the Docker socket is not mounted inside any containers (Scored)

honeur/zeppelin:0.8.2-2.0.0

Command

```
git clone https://github.com/docker/docker-bench-security.git

cd docker-bench-security
```

```
sudo sh docker-bench-security.sh -i honeur/zeppelin:0.8.2-2.0.0,zeppelin -c
container_images,container_runtime
```

Output

[INFO] 4 - Container Images and Build File
[WARN] 4.1 - Ensure that a user for the container has been created (Scored)
[WARN]   * Running as root: zeppelin
[NOTE] 4.2 - Ensure that containers use only trusted base images (Not Scored)
[NOTE] 4.3 - Ensure that unnecessary packages are not installed in the container (Not Scored)
[NOTE] 4.4 - Ensure images are scanned and rebuilt to include security patches (Not Scored)
[WARN] 4.5 - Ensure Content trust for Docker is Enabled (Scored)
[PASS] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images (Scored)
[INFO] 4.7 - Ensure update instructions are not use alone in the Dockerfile (Not Scored)
[INFO]   * Update instruction found: [honeur/zeppelin:0.8.2-2.0.0]
[NOTE] 4.8 - Ensure setuid and setgid permissions are removed (Not Scored)
[PASS] 4.9 - Ensure that COPY is used instead of ADD in Dockerfiles (Not Scored)
[NOTE] 4.10 - Ensure secrets are not stored in Dockerfiles (Not Scored)
[NOTE] 4.11 - Ensure only verified packages are are installed (Not Scored)

[INFO] 5 - Container Runtime
[WARN] 5.1 - Ensure that, if applicable, an AppArmor Profile is enabled (Scored)
[WARN]   * No AppArmorProfile Found: zeppelin
[PASS] 5.2 - Ensure that, if applicable, SELinux security options are set (Scored)
[PASS] 5.3 - Ensure that Linux kernel capabilities are restricted within containers (Scored)
[PASS] 5.4 - Ensure that privileged containers are not used (Scored)
[PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers (Scored)
[PASS] 5.6 - Ensure sshd is not run within containers (Scored)
[PASS] 5.7 - Ensure privileged ports are not mapped within containers (Scored)
[NOTE] 5.8 - Ensure that only needed ports are open on the container (Not Scored)
[PASS] 5.9 - Ensure that the host's network namespace is not shared (Scored)
[PASS] 5.10 - Ensure that the memory usage for containers is limited (Scored)
[PASS] 5.11 - Ensure that CPU priority is set appropriately on containers (Scored)
[WARN] 5.12 - Ensure that the container's root filesystem is mounted as read only (Scored)
[WARN]   * Container running with root FS mounted R/W: zeppelin
[PASS] 5.13 - Ensure that incoming container traffic is bound to a specific host interface (Scored)
[PASS] 5.14 - Ensure that the 'on-failure' container restart policy is set to '5' (Scored)
[PASS] 5.15 - Ensure that the host's process namespace is not shared (Scored)
[PASS] 5.16 - Ensure that the host's IPC namespace is not shared (Scored)
[PASS] 5.17 - Ensure that host devices are not directly exposed to containers (Not Scored)
[PASS] 5.18 - Ensure that the default ulimit is overwritten at runtime if needed (Not Scored)
[PASS] 5.19 - Ensure mount propagation mode is not set to shared (Scored)
[PASS] 5.20 - Ensure that the host's UTS namespace is not shared (Scored)
[PASS] 5.21 - Ensure the default seccomp profile is not Disabled (Scored)
[NOTE] 5.22 - Ensure that docker exec commands are not used with the privileged option (Scored)
[NOTE] 5.23 - Ensure that docker exec commands are not used with the user=root option (Not Scored)
[PASS] 5.24 - Ensure that cgroup usage is confirmed (Scored)
[PASS] 5.25 - Ensure that the container is restricted from acquiring additional privileges (Scored)
[PASS] 5.26 - Ensure that container health is checked at runtime (Scored)
[INFO] 5.27 - Ensure that Docker commands always make use of the latest version of their image (Not Scored)
[PASS] 5.28 - Ensure that the PIDs cgroup limit is used (Scored)
[INFO] 5.29 - Ensure that Docker's default bridge docker0 is not used (Not Scored)
[INFO]   * Container in docker0 network: zeppelin
[PASS] 5.30 - Ensure that the host's user namespaces are not shared (Scored)
[PASS] 5.31 - Ensure that the Docker socket is not mounted inside any containers (Scored)

honeur/user-mgmt:2.0.0

Command

```
git clone https://github.com/docker/docker-bench-security.git
cd docker-bench-security
sudo sh docker-bench-security.sh -i honeur/user-mgmt:2.0.0,user-mgmt -c
container_images,container_runtime
```

Output

[INFO] 4 - Container Images and Build File
[PASS] 4.1 - Ensure that a user for the container has been created (Scored)
[NOTE] 4.2 - Ensure that containers use only trusted base images (Not Scored)
[NOTE] 4.3 - Ensure that unnecessary packages are not installed in the container (Not Scored)
[NOTE] 4.4 - Ensure images are scanned and rebuilt to include security patches (Not Scored)
[WARN] 4.5 - Ensure Content trust for Docker is Enabled (Scored)
[PASS] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images (Scored)
[INFO] 4.7 - Ensure update instructions are not use alone in the Dockerfile (Not Scored)
[INFO]   * Update instruction found: [honeur/user-mgmt:2.0.0]
[NOTE] 4.8 - Ensure setuid and setgid permissions are removed (Not Scored)
[PASS] 4.9 - Ensure that COPY is used instead of ADD in Dockerfiles (Not Scored)
[NOTE] 4.10 - Ensure secrets are not stored in Dockerfiles (Not Scored)
[NOTE] 4.11 - Ensure only verified packages are are installed (Not Scored)

[INFO] 5 - Container Runtime
[WARN] 5.1 - Ensure that, if applicable, an AppArmor Profile is enabled (Scored)
[WARN]   * No AppArmorProfile Found: user-mgmt
[PASS] 5.2 - Ensure that, if applicable, SELinux security options are set (Scored)
[PASS] 5.3 - Ensure that Linux kernel capabilities are restricted within containers (Scored)
[PASS] 5.4 - Ensure that privileged containers are not used (Scored)
[PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers (Scored)
[PASS] 5.6 - Ensure sshd is not run within containers (Scored)
[PASS] 5.7 - Ensure privileged ports are not mapped within containers (Scored)
[NOTE] 5.8 - Ensure that only needed ports are open on the container (Not Scored)
[PASS] 5.9 - Ensure that the host's network namespace is not shared (Scored)
[PASS] 5.10 - Ensure that the memory usage for containers is limited (Scored)
[PASS] 5.11 - Ensure that CPU priority is set appropriately on containers (Scored)
[WARN] 5.12 - Ensure that the container's root filesystem is mounted as read only (Scored)
[WARN]   * Container running with root FS mounted R/W: user-mgmt
[PASS] 5.13 - Ensure that incoming container traffic is bound to a specific host interface (Scored)
[PASS] 5.14 - Ensure that the 'on-failure' container restart policy is set to '5' (Scored)
[PASS] 5.15 - Ensure that the host's process namespace is not shared (Scored)
[PASS] 5.16 - Ensure that the host's IPC namespace is not shared (Scored)
[PASS] 5.17 - Ensure that host devices are not directly exposed to containers (Not Scored)
[PASS] 5.18 - Ensure that the default ulimit is overwritten at runtime if needed (Not Scored)
[PASS] 5.19 - Ensure mount propagation mode is not set to shared (Scored)
[PASS] 5.20 - Ensure that the host's UTS namespace is not shared (Scored)
[PASS] 5.21 - Ensure the default seccomp profile is not Disabled (Scored)
[NOTE] 5.22 - Ensure that docker exec commands are not used with the privileged option (Scored)
[NOTE] 5.23 - Ensure that docker exec commands are not used with the user=root option (Not Scored)
[PASS] 5.24 - Ensure that cgroup usage is confirmed (Scored)
[PASS] 5.25 - Ensure that the container is restricted from acquiring additional privileges (Scored)
[PASS] 5.26 - Ensure that container health is checked at runtime (Scored)
[INFO] 5.27 - Ensure that Docker commands always make use of the latest version of their image (Not Scored)
[PASS] 5.28 - Ensure that the PIDs cgroup limit is used (Scored)
[INFO] 5.29 - Ensure that Docker's default bridge docker0 is not used (Not Scored)

[INFO]   * Container in docker0 network: user-mgmt
[PASS] 5.30 - Ensure that the host's user namespaces are not shared (Scored)
[PASS] 5.31 - Ensure that the Docker socket is not mounted inside any containers (Scored)

honeur/distributed-analytice:remote-2.0.0

Command

```
git clone https://github.com/docker/docker-bench-security.git

cd docker-bench-security

sudo sh docker-bench-security.sh -i

honeur/distributed-analytics:remote-2.0.0,distributed-analytics-remote -c

container_images,container_runtime
```

Output

[INFO] 4 - Container Images and Build File
[PASS] 4.1 - Ensure that a user for the container has been created (Scored)
[NOTE] 4.2 - Ensure that containers use only trusted base images (Not Scored)
[NOTE] 4.3 - Ensure that unnecessary packages are not installed in the container (Not Scored)
[NOTE] 4.4 - Ensure images are scanned and rebuilt to include security patches (Not Scored)
[WARN] 4.5 - Ensure Content trust for Docker is Enabled (Scored)
[PASS] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images (Scored)
[PASS] 4.7 - Ensure update instructions are not use alone in the Dockerfile (Not Scored)
[NOTE] 4.8 - Ensure setuid and setgid permissions are removed (Not Scored)
[PASS] 4.9 - Ensure that COPY is used instead of ADD in Dockerfiles (Not Scored)
[NOTE] 4.10 - Ensure secrets are not stored in Dockerfiles (Not Scored)
[NOTE] 4.11 - Ensure only verified packages are are installed (Not Scored)

[INFO] 5 - Container Runtime
[WARN] 5.1 - Ensure that, if applicable, an AppArmor Profile is enabled (Scored)
[WARN]   * No AppArmorProfile Found: distributed-analytics-remote
[PASS] 5.2 - Ensure that, if applicable, SELinux security options are set (Scored)
[PASS] 5.3 - Ensure that Linux kernel capabilities are restricted within containers (Scored)
[PASS] 5.4 - Ensure that privileged containers are not used (Scored)
[PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers (Scored)
[PASS] 5.6 - Ensure sshd is not run within containers (Scored)
[PASS] 5.7 - Ensure privileged ports are not mapped within containers (Scored)
[NOTE] 5.8 - Ensure that only needed ports are open on the container (Not Scored)
[PASS] 5.9 - Ensure that the host's network namespace is not shared (Scored)
[PASS] 5.10 - Ensure that the memory usage for containers is limited (Scored)
[PASS] 5.11 - Ensure that CPU priority is set appropriately on containers (Scored)
[WARN] 5.12 - Ensure that the container's root filesystem is mounted as read only (Scored)
[WARN]   * Container running with root FS mounted R/W: distributed-analytics-remote
[PASS] 5.13 - Ensure that incoming container traffic is bound to a specific host interface (Scored)
[PASS] 5.14 - Ensure that the 'on-failure' container restart policy is set to '5' (Scored)
[PASS] 5.15 - Ensure that the host's process namespace is not shared (Scored)
[PASS] 5.16 - Ensure that the host's IPC namespace is not shared (Scored)
[PASS] 5.17 - Ensure that host devices are not directly exposed to containers (Not Scored)
[PASS] 5.18 - Ensure that the default ulimit is overwritten at runtime if needed (Not Scored)
[PASS] 5.19 - Ensure mount propagation mode is not set to shared (Scored)
[PASS] 5.20 - Ensure that the host's UTS namespace is not shared (Scored)
[PASS] 5.21 - Ensure the default seccomp profile is not Disabled (Scored)
[NOTE] 5.22 - Ensure that docker exec commands are not used with the privileged option (Scored)
[NOTE] 5.23 - Ensure that docker exec commands are not used with the user=root option (Not Scored)

[PASS] 5.24 - Ensure that cgroup usage is confirmed (Scored)
[PASS] 5.25 - Ensure that the container is restricted from acquiring additional privileges (Scored)
[PASS] 5.26 - Ensure that container health is checked at runtime (Scored)
[INFO] 5.27 - Ensure that Docker commands always make use of the latest version of their image (Not Scored)
[PASS] 5.28 - Ensure that the PIDs cgroup limit is used (Scored)
[INFO] 5.29 - Ensure that Docker's default bridge docker0 is not used (Not Scored)
[INFO]   * Container in docker0 network: distributed-analytics-remote
[PASS] 5.30 - Ensure that the host's user namespaces are not shared (Scored)
[PASS] 5.31 - Ensure that the Docker socket is not mounted inside any containers (Scored)

honeur/distributed-analytice:r-server-2.0.0

Command

```
git clone https://github.com/docker/docker-bench-security.git

cd docker-bench-security

sudo sh docker-bench-security.sh -i

honeur/distributed-analytics:r-server-2.0.0,distributed-analytics-r-server -c

container_images,container_runtime
```

Output

[INFO] 4 - Container Images and Build File
[WARN] 4.1 - Ensure that a user for the container has been created (Scored)
[WARN]   * Running as root: distributed-analytics-r-server
[NOTE] 4.2 - Ensure that containers use only trusted base images (Not Scored)
[NOTE] 4.3 - Ensure that unnecessary packages are not installed in the container (Not Scored)
[NOTE] 4.4 - Ensure images are scanned and rebuilt to include security patches (Not Scored)
[WARN] 4.5 - Ensure Content trust for Docker is Enabled (Scored)
[PASS] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images (Scored)
[PASS] 4.7 - Ensure update instructions are not use alone in the Dockerfile (Not Scored)
[NOTE] 4.8 - Ensure setuid and setgid permissions are removed (Not Scored)
[PASS] 4.9 - Ensure that COPY is used instead of ADD in Dockerfiles (Not Scored)
[NOTE] 4.10 - Ensure secrets are not stored in Dockerfiles (Not Scored)
[NOTE] 4.11 - Ensure only verified packages are are installed (Not Scored)

[INFO] 5 - Container Runtime
[WARN] 5.1 - Ensure that, if applicable, an AppArmor Profile is enabled (Scored)
[WARN]   * No AppArmorProfile Found: distributed-analytics-r-server
[PASS] 5.2 - Ensure that, if applicable, SELinux security options are set (Scored)
[PASS] 5.3 - Ensure that Linux kernel capabilities are restricted within containers (Scored)
[PASS] 5.4 - Ensure that privileged containers are not used (Scored)
[PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers (Scored)
[PASS] 5.6 - Ensure sshd is not run within containers (Scored)
[PASS] 5.7 - Ensure privileged ports are not mapped within containers (Scored)
[NOTE] 5.8 - Ensure that only needed ports are open on the container (Not Scored)
[PASS] 5.9 - Ensure that the host's network namespace is not shared (Scored)
[PASS] 5.10 - Ensure that the memory usage for containers is limited (Scored)
[PASS] 5.11 - Ensure that CPU priority is set appropriately on containers (Scored)
[WARN] 5.12 - Ensure that the container's root filesystem is mounted as read only (Scored)
[WARN]   * Container running with root FS mounted R/W: distributed-analytics-r-server
[PASS] 5.13 - Ensure that incoming container traffic is bound to a specific host interface (Scored)
[PASS] 5.14 - Ensure that the 'on-failure' container restart policy is set to '5' (Scored)
[PASS] 5.15 - Ensure that the host's process namespace is not shared (Scored)
[PASS] 5.16 - Ensure that the host's IPC namespace is not shared (Scored)

[PASS] 5.17 - Ensure that host devices are not directly exposed to containers (Not Scored)
[PASS] 5.18 - Ensure that the default ulimit is overwritten at runtime if needed (Not Scored)
[PASS] 5.19 - Ensure mount propagation mode is not set to shared (Scored)
[PASS] 5.20 - Ensure that the host's UTS namespace is not shared (Scored)
[PASS] 5.21 - Ensure the default seccomp profile is not Disabled (Scored)
[NOTE] 5.22 - Ensure that docker exec commands are not used with the privileged option (Scored)
[NOTE] 5.23 - Ensure that docker exec commands are not used with the user=root option (Not Scored)
[PASS] 5.24 - Ensure that cgroup usage is confirmed (Scored)
[PASS] 5.25 - Ensure that the container is restricted from acquiring additional privileges (Scored)
[PASS] 5.26 - Ensure that container health is checked at runtime (Scored)
[INFO] 5.27 - Ensure that Docker commands always make use of the latest version of their image (Not Scored)
[PASS] 5.28 - Ensure that the PIDs cgroup limit is used (Scored)
[INFO] 5.29 - Ensure that Docker's default bridge docker0 is not used (Not Scored)
[INFO] * Container in docker0 network: distributed-analytics-r-server
[PASS] 5.30 - Ensure that the host's user namespaces are not shared (Scored)
[PASS] 5.31 - Ensure that the Docker socket is not mounted inside any containers (Scored)

honeur/honeur-studio:2.0.0

Command

```
git clone https://github.com/docker/docker-bench-security.git
cd docker-bench-security
sudo sh docker-bench-security.sh -i
honeur/honeur-studio:2.0.0,honeur-studio,honeur-studio-chronicle,<vscode>,<rstudio>,<reports>,<documents>,<personal> -c container_images,container_runtime
```

Output

[INFO] 4 - Container Images and Build File
[WARN] 4.1 - Ensure that a user for the container has been created (Scored)
[WARN] * Running as root: personal
[WARN] * Running as root: documents
[WARN] * Running as root: reports
[WARN] * Running as root: rstudio
[WARN] * Running as root: vscode
[WARN] * Running as root: honeur-studio
[WARN] * Running as root: honeur-studio-chronicle
[NOTE] 4.2 - Ensure that containers use only trusted base images (Not Scored)
[NOTE] 4.3 - Ensure that unnecessary packages are not installed in the container (Not Scored)
[NOTE] 4.4 - Ensure images are scanned and rebuilt to include security patches (Not Scored)
[WARN] 4.5 - Ensure Content trust for Docker is Enabled (Scored)
[WARN] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images (Scored)
[WARN] * No Healthcheck found: [honeur/honeur-studio:2.0.0]
[PASS] 4.7 - Ensure update instructions are not use alone in the Dockerfile (Not Scored)
[NOTE] 4.8 - Ensure setuid and setgid permissions are removed (Not Scored)
[PASS] 4.9 - Ensure that COPY is used instead of ADD in Dockerfiles (Not Scored)
[NOTE] 4.10 - Ensure secrets are not stored in Dockerfiles (Not Scored)
[NOTE] 4.11 - Ensure only verified packages are are installed (Not Scored)

[INFO] 5 - Container Runtime
[WARN] 5.1 - Ensure that, if applicable, an AppArmor Profile is enabled (Scored)
[WARN] * No AppArmorProfile Found: personal
[WARN] * No AppArmorProfile Found: documents
[WARN] * No AppArmorProfile Found: reports

[WARN]   * No AppArmorProfile Found: rstudio
[WARN]   * No AppArmorProfile Found: vscode
[WARN]   * No AppArmorProfile Found: honeur-studio
[WARN]   * No AppArmorProfile Found: honeur-studio-chronicle
[WARN] 5.2 - Ensure that, if applicable, SELinux security options are set (Scored)
[WARN]   * No SecurityOptions Found: personal
[WARN]   * No SecurityOptions Found: documents
[WARN]   * No SecurityOptions Found: reports
[WARN]   * No SecurityOptions Found: rstudio
[WARN]   * No SecurityOptions Found: vscode
[PASS] 5.3 - Ensure that Linux kernel capabilities are restricted within containers (Scored)
[PASS] 5.4 - Ensure that privileged containers are not used (Scored)
[PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers (Scored)
[PASS] 5.6 - Ensure sshd is not run within containers (Scored)
[PASS] 5.7 - Ensure privileged ports are not mapped within containers (Scored)
[NOTE] 5.8 - Ensure that only needed ports are open on the container (Not Scored)
[PASS] 5.9 - Ensure that the host's network namespace is not shared (Scored)
[WARN] 5.10 - Ensure that the memory usage for containers is limited (Scored)
[WARN]   * Container running without memory restrictions: personal
[WARN]   * Container running without memory restrictions: documents
[WARN]   * Container running without memory restrictions: reports
[WARN]   * Container running without memory restrictions: rstudio
[WARN]   * Container running without memory restrictions: vscode
[WARN] 5.11 - Ensure that CPU priority is set appropriately on containers (Scored)
[WARN]   * Container running without CPU restrictions: personal
[WARN]   * Container running without CPU restrictions: documents
[WARN]   * Container running without CPU restrictions: reports
[WARN]   * Container running without CPU restrictions: rstudio
[WARN]   * Container running without CPU restrictions: vscode
[WARN] 5.12 - Ensure that the container's root filesystem is mounted as read only (Scored)
[WARN]   * Container running with root FS mounted R/W: personal
[WARN]   * Container running with root FS mounted R/W: documents
[WARN]   * Container running with root FS mounted R/W: reports
[WARN]   * Container running with root FS mounted R/W: rstudio
[WARN]   * Container running with root FS mounted R/W: vscode
[WARN]   * Container running with root FS mounted R/W: honeur-studio
[WARN]   * Container running with root FS mounted R/W: honeur-studio-chronicle
[PASS] 5.13 - Ensure that incoming container traffic is bound to a specific host interface (Scored)
[WARN] 5.14 - Ensure that the 'on-failure' container restart policy is set to '5' (Scored)
[WARN]   * MaximumRetryCount is not set to 5: personal
[WARN]   * MaximumRetryCount is not set to 5: documents
[WARN]   * MaximumRetryCount is not set to 5: reports
[WARN]   * MaximumRetryCount is not set to 5: rstudio
[WARN]   * MaximumRetryCount is not set to 5: vscode
[PASS] 5.15 - Ensure that the host's process namespace is not shared (Scored)
[PASS] 5.16 - Ensure that the host's IPC namespace is not shared (Scored)
[PASS] 5.17 - Ensure that host devices are not directly exposed to containers (Not Scored)
[INFO] 5.18 - Ensure that the default ulimit is overwritten at runtime if needed (Not Scored)
[INFO]   * Container no default ulimit override: personal
[INFO]   * Container no default ulimit override: documents
[INFO]   * Container no default ulimit override: reports
[INFO]   * Container no default ulimit override: rstudio
[INFO]   * Container no default ulimit override: vscode
[PASS] 5.19 - Ensure mount propagation mode is not set to shared (Scored)
[PASS] 5.20 - Ensure that the host's UTS namespace is not shared (Scored)
[PASS] 5.21 - Ensure the default seccomp profile is not Disabled (Scored)
[NOTE] 5.22 - Ensure that docker exec commands are not used with the privileged option (Scored)
[NOTE] 5.23 - Ensure that docker exec commands are not used with the user=root option (Not Scored)
[PASS] 5.24 - Ensure that cgroup usage is confirmed (Scored)
[WARN] 5.25 - Ensure that the container is restricted from acquiring additional privileges (Scored)
[WARN]   * Privileges not restricted: personal
[WARN]   * Privileges not restricted: documents

[WARN]   * Privileges not restricted: reports
[WARN]   * Privileges not restricted: rstudio
[WARN]   * Privileges not restricted: vscode
[WARN] 5.26 - Ensure that container health is checked at runtime (Scored)
[WARN]   * Health check not set: personal
[WARN]   * Health check not set: documents
[WARN]   * Health check not set: reports
[WARN]   * Health check not set: rstudio
[WARN]   * Health check not set: vscode
[WARN]   * Health check not set: honeur-studio
[WARN]   * Health check not set: honeur-studio-chronicle
[INFO] 5.27 - Ensure that Docker commands always make use of the latest version of their image (Not Scored)
[WARN] 5.28 - Ensure that the PIDs cgroup limit is used (Scored)
[WARN]   * PIDs limit not set: personal
[WARN]   * PIDs limit not set: documents
[WARN]   * PIDs limit not set: reports
[WARN]   * PIDs limit not set: rstudio
[WARN]   * PIDs limit not set: vscode
[INFO] 5.29 - Ensure that Docker's default bridge docker0 is not used (Not Scored)
[INFO]   * Container in docker0 network: personal
[INFO]   * Container in docker0 network: reports
[INFO]   * Container in docker0 network: documents
[INFO]   * Container in docker0 network: vscode
[INFO]   * Container in docker0 network: rstudio
[INFO]   * Container in docker0 network: honeur-studio
[PASS] 5.30 - Ensure that the host's user namespaces are not shared (Scored)
[WARN] 5.31 - Ensure that the Docker socket is not mounted inside any containers (Scored)
[WARN]   * Docker socket shared: honeur-studio

honeur/nginx:2.0.0

Command

```
git clone https://github.com/docker/docker-bench-security.git

cd docker-bench-security

sudo sh docker-bench-security.sh -i honeur/nginx:2.0.0,nginx -c

container_images,container_runtime
```

Output

[INFO] 4 - Container Images and Build File
[WARN] 4.1 - Ensure that a user for the container has been created (Scored)
[WARN]   * Running as root: nginx
[NOTE] 4.2 - Ensure that containers use only trusted base images (Not Scored)
[NOTE] 4.3 - Ensure that unnecessary packages are not installed in the container (Not Scored)
[NOTE] 4.4 - Ensure images are scanned and rebuilt to include security patches (Not Scored)
[WARN] 4.5 - Ensure Content trust for Docker is Enabled (Scored)
[WARN] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images (Scored)
[WARN]   * No Healthcheck found: [honeur/nginx:2.0.0]
[PASS] 4.7 - Ensure update instructions are not use alone in the Dockerfile (Not Scored)
[NOTE] 4.8 - Ensure setuid and setgid permissions are removed (Not Scored)
[PASS] 4.9 - Ensure that COPY is used instead of ADD in Dockerfiles (Not Scored)
[NOTE] 4.10 - Ensure secrets are not stored in Dockerfiles (Not Scored)
[NOTE] 4.11 - Ensure only verified packages are are installed (Not Scored)

[INFO] 5 - Container Runtime
[WARN] 5.1 - Ensure that, if applicable, an AppArmor Profile is enabled (Scored)

[WARN]   * No AppArmorProfile Found: nginx
[PASS] 5.2 - Ensure that, if applicable, SELinux security options are set (Scored)
[PASS] 5.3 - Ensure that Linux kernel capabilities are restricted within containers (Scored)
[PASS] 5.4 - Ensure that privileged containers are not used (Scored)
[PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers (Scored)
[PASS] 5.6 - Ensure sshd is not run within containers (Scored)
[WARN] 5.7 - Ensure privileged ports are not mapped within containers (Scored)
[WARN]   * Privileged Port in use: 80 in nginx
[NOTE] 5.8 - Ensure that only needed ports are open on the container (Not Scored)
[PASS] 5.9 - Ensure that the host's network namespace is not shared (Scored)
[PASS] 5.10 - Ensure that the memory usage for containers is limited (Scored)
[PASS] 5.11 - Ensure that CPU priority is set appropriately on containers (Scored)
[WARN] 5.12 - Ensure that the container's root filesystem is mounted as read only (Scored)
[WARN]   * Container running with root FS mounted R/W: nginx
[WARN] 5.13 - Ensure that incoming container traffic is bound to a specific host interface (Scored)
[WARN]   * Port being bound to wildcard IP: 0.0.0.0 in nginx
[PASS] 5.14 - Ensure that the 'on-failure' container restart policy is set to '5' (Scored)
[PASS] 5.15 - Ensure that the host's process namespace is not shared (Scored)
[PASS] 5.16 - Ensure that the host's IPC namespace is not shared (Scored)
[PASS] 5.17 - Ensure that host devices are not directly exposed to containers (Not Scored)
[PASS] 5.18 - Ensure that the default ulimit is overwritten at runtime if needed (Not Scored)
[PASS] 5.19 - Ensure mount propagation mode is not set to shared (Scored)
[PASS] 5.20 - Ensure that the host's UTS namespace is not shared (Scored)
[PASS] 5.21 - Ensure the default seccomp profile is not Disabled (Scored)
[NOTE] 5.22 - Ensure that docker exec commands are not used with the privileged option (Scored)
[NOTE] 5.23 - Ensure that docker exec commands are not used with the user=root option (Not Scored)
[PASS] 5.24 - Ensure that cgroup usage is confirmed (Scored)
[PASS] 5.25 - Ensure that the container is restricted from acquiring additional privileges (Scored)
[WARN] 5.26 - Ensure that container health is checked at runtime (Scored)
[WARN]   * Health check not set: nginx
[INFO] 5.27 - Ensure that Docker commands always make use of the latest version of their image (Not Scored)
[PASS] 5.28 - Ensure that the PIDs cgroup limit is used (Scored)
[PASS] 5.29 - Ensure that Docker's default bridge docker0 is not used (Not Scored)
[PASS] 5.30 - Ensure that the host's user namespaces are not shared (Scored)
[PASS] 5.31 - Ensure that the Docker socket is not mounted inside any containers (Scored)

Clarifications

WARNINGS
[WARN] 4.1 - Ensure a user for the container has been created
The final command or entrypoint of an image should never be run as a root user. If the container should get compromised, root access is granted to the attacker which is never good.

honeur/honeur-distributed-analytics:r-server-2.0.0 start as root to change the permissions of the shared volume between host, r-server and zeppelin. Once the permissions of the shared volume are changed, the r-server is run by the **nobody** user.
honeur/zeppelin:0.8.2-2.0.0 we used apache:zeppelin:0.8.2 as the base image and only did configuration changes. Because the base image has a few programs depending on root access in the container, we decided to keep the root user.
honeur/honeur-studio:2.0.0 starts the container as root but it uses the S6 overlay docker integration so after changing permissions it changes to a predefined non-root user.
honeur/nginx:2.0.0 needs root access to open up port 80. This is a system port and can not be opened without root access.
[WARN] 4.5 - Ensure Content trust for Docker is Enabled
You can enable content trust in docker setting the environment variable DOCKER_CONTENT_TRUST to 1:
export DOCKER_CONTENT_TRUST=1
This will tell docker that it only should trust signed images.
All the images delivered by HONEUR are signed.
[WARN] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images (Scored)
honeur/honeur-studio:2.0.0 packs all the different sub-applications like vscode, rstudio, shinyproxy,... Depending on the entrypoint and command it will run a different application so no standard health check could be created.
honeur/nginx:2.0.0 health is indirectly checked by accessing localhost or the hostname of the host machine where HONEUR Components are build on.
[WARN] 5.1 - Ensure that, if applicable, an AppArmor Profile is enabled (Scored)

This are runtime warnings and are not related to the HONEUR images. AppArmor profiles can be created for more security by the remote sites for each HONEUR Component.

[WARN] 5.2 - Ensure that, if applicable, SELinux security options are set (Scored)

Only on honeur-studio sub-application like vscode, rstudio, shinyproxy,... no security options are available. This is because the creation of honeur-studio sub-applications containers are controlled by the parent honeur-studio container.

[WARN] 5.10 - Ensure that the memory usage for containers is limited

This are runtime warnings and are not related to the HONEUR images.   CGroups can be created on the host machine and referenced in the docker containers to avoid this warning.

[WARN] 5.11 - Ensure CPU priority is set appropriately on the container

This are runtime warnings and are not related to the HONEUR images.   CGroups can be created on the host machine and referenced in the docker containers to avoid this warning.

[WARN] 5.12 - Ensure that the container's root filesystem is mounted as read only

Some services on operating systems still work with access to the root filesystem. In our case the root filesystem was used to write process locks to. If we enabled read only for the root filesystem, we could not get the main programs run in the containers.

[WARN] 5.13 - Ensure that incoming container traffic is bound to a specific host interface

0.0.0.0 is a wildcard IP Address. We opted for this instead of the host IP address because our remote system will mostly run on computers with dynamic IP Addresses. It is still possible to change this for servers with a fixed IP address.

[WARN] 5.14 - Ensure that the 'on-failure' container restart policy is set to '5'

The restart policy is set to none on the honeur-studio sub-application containers. These containers are controlled by the parent honeur-studio container and will kill the container if a certain threshold is met or when failure is due.

[WARN] 5.26 - Ensure that container health is checked at runtime (Scored)

same as warning 4.6

[WARN] 5.28 - Ensure that the PIDs cgroup limit is used

This are runtime warnings and are not directly related to the HONEUR images. CGroups can be created on host machine and referenced in the docker container to avoid this warning.

[WARN] 5.31 - Ensure that the Docker socket is not mounted inside any containers (Scored)

honeur-studio parent container does need the docker socket for creating additional containers like vscode, rstudio, shiny,... during runtime. This is an integral part of the inner workings of honeur-studio.