

CAST AIP Docker Installer - Linux installation - start_docker_dashboards.sh

- Run the installer
- Choose simple or advanced installation
 - Simple mode
 - CAST Storage Service/PostgreSQL parameters
 - Additional details
 - Advanced mode
 - Dashboard and port number selection
 - CAST Storage Service/PostgreSQL parameters
 - Additional details
 - Data persistence requirements
- Check summary
- Check container and image
- Test Dashboard access
 - Default configuration
- Modifying the default configuration
 - Simple mode choice
 - Advanced mode choice with data stored outside the container

Run the installer

Run the Linux installer script with the following command:

```
./start_docker_dashboards.sh
```



Note that you may need to run "`chmod +x ./start_docker_dashboards.sh`" to grant execution permission to the script before running it. You may also need to run this file with elevated permissions (e.g. `sudo`).

Choose simple or advanced installation

The installer will run and offer you a choice between **simple** or **advanced** install mode:


```

Enter the host value for CSS server:
192.168.200.104
The host 192.168.200.104 will be used as CSS host.

Enter the port value for CSS server:
2282

Enter the value for CSS user, (default is "operator"):

Enter the value for CSS password (default is "CastAIP"):

```

hostname	Enter the hostname/IP address and port number of the CAST Storage Service/PostgreSQL instance on which the AIP schemas are stored. This CAST Storage Service/PostgreSQL instance can either be installed on the local server or a remote server.
port	<div style="border: 1px solid #ccc; padding: 5px;"> <p>i hostname must not be localhost even if the CAST Storage Service/PostgreSQL instance is located on the same server as your Docker installation. Keep in mind that a Docker container is a standalone package of software, similar to a lightweight virtual machine. Thus, using localhost will be seen by the container as a server hosted within the container itself and not as the server on which Docker is installed.</p> </div>
user	Enter the credentials for the CAST Storage Service/PostgreSQL instance. The default options are operator/CastAIP .
password	

Additional details

```

=====
Now, talking about the schemas to use from your CSS server
Enter the name of your measurement schema:
general_measure_125
Enter the names (a comma separated list) of your central schemas:
meudon_01_central
Enter the CAST Dashboard license:
[REDACTED]
Writing schemas and license in the config file.
Writing central schema and license in the config file.

```

Name of your Measure schema	Enter the name of your Measure schema that will be used by the Health Dashboard.
Names of your Dashboard schemas	Enter name(s) of your Dashboard schemas for use with the Engineering or Security Dashboard. If you have multiple schemas to use, enter them separated by a comma, for example: schema1_central,schema2_central,schema3_central
License key for access to the Dashboard schemas	Enter the license key for access to your Dashboard schemas via the Engineering or Security Dashboard. See Dashboard Service license key configuration .

In a new installation, no existing containers will be detected, and the installer will then create a new Docker container with the default name **castdashboard_s_1**. You can enter a custom name if you prefer:

```

No old container to clean.
Container name could be: << castdashboards_1 >>, if you want a custom one, enter its name or Return if this one is ok for you:

```

The installer will then start the process of pulling the Docker image:

```
=====
Updating current image with latest one:
latest: Pulling from castaip/dashboards_linux
e7c96db7181b: Pull complete
f910a506b6cb: Pull complete
b6abafe80f63: Downloading [=====>] 31.06MB/54.94MB
391185a84357: Download complete
ae25fbce7c44: Downloading [=====>] 19.26MB/104MB
de2821164edb: Downloading [=====>] 19.38MB/90.54MB
cae7e1b50983: Waiting
2527d7890269: Waiting
586287663e34: Waiting
```

The container will then start:

```
=====
Starting a new container with ID:
Docker command: docker run -d -v /tmp/cast_dashboards_docker:/workspace --name castdashboards_1 -p 8080:8080 -p 8081:8081 -p 8082:8082 -p 8083:8083 castaip/dashboards_linux:latest
```

 The items are all downloaded directly from <https://hub.docker.com/>.

Advanced mode

In **Advanced mode**, you will be prompted to select

- the Dashboards that you want to start and choose their port numbers
- the parameters of your CAST Storage Service/PostgreSQL instance on which the AIP schemas you want to use with the CAST Dashboards are stored

Dashboard and port number selection

```

=====
Advanced mode.
=====
=====

Make a choice about which dashboards you want to use:

-- Health & Engineering Dashboard: (default=y)
  Make a choice: y / n

Enter the port value for CAST Health and Engineering Dashboard: (default is: 8080)

-- CAST Health Dashboard: (default=y)
  Make a choice: y / n

Enter the port value for CAST Health Dashboard: (default is: 8081)

-- CAST Engineering Dashboard: (default=y)
  Make a choice: y / n

Enter the port value for CAST Engineering Dashboard: (default is: 8082)

-- CAST Security Dashboard: (default=y)
  Make a choice: y / n

Enter the port value for CAST Security Dashboard: (default is: 8083)

```

I

Health and Engineering Dashboard	Choose to install both the Health and Engineering Dashboard as a "combined" installation. The default port number will be set to 8080 , but you can choose another custom port if required.
Health Dashboard	Choose to install the Health Dashboard alone. The default port number will be set to 8081 , but you can choose another custom port if required.
Engineering Dashboard	Choose to install the Engineering Dashboard alone. The default port number will be set to 8082 , but you can choose another custom port if required.
Security Dashboard	Choose to install the Security Dashboard alone. The default port number will be set to 8083 , but you can choose another custom port if required.

CAST Storage Service/PostgreSQL parameters

```

Enter the host value for CSS server:
192.168.200.104
The host 192.168.200.104 will be used as CSS host.

Enter the port value for CSS server:
2282

Enter the value for CSS user, (default is "operator"):

Enter the value for CSS password (default is "CastAIP"):

```

hostname	Enter the hostname/IP address and port number of the CAST Storage Service/PostgreSQL instance on which the AIP schemas are stored. This CAST Storage Service/PostgreSQL instance can either be installed on the local server or a remote server.
port	<div style="border: 1px solid #ccc; padding: 5px;"> <p>i hostname must not be localhost even if the CAST Storage Service/PostgreSQL instance is located on the same server as your Docker installation. Keep in mind that a Docker container is a standalone package of software, similar to a lightweight virtual machine. Thus, using localhost will be seen by the container as a server hosted within the container itself and not as the server on which Docker is installed.</p> </div>
user	Enter the credentials for the CAST Storage Service/PostgreSQL instance. The default options are operator/CastAIP .
password	

Additional details

```

=====
Now, talking about the schemas to use from your CSS server
Enter the name of your measurement schema:
general_measure_125
Enter the names (a comma separated list) of your central schemas:
meudon_01_central
Enter the CAST Dashboard license:
_____
Writing schemas and license in the config file.
Writing central schema and license in the config file.

```

Name of your Measure schema	Enter the name of your Measure schema that will be used by the Health Dashboard - if you have chosen to install it.
Names of your Dashboard schemas	Enter name(s) of your Dashboard schemas for use with the Engineering or Security Dashboard - if you have chosen to install them. If you have multiple schemas to use, enter them separated by a comma, for example: schema1_central,schema2_central,schema3_central
License key for access to the Dashboard schemas	Enter the license key for access to your Dashboard schemas via the Engineering or Security Dashboard. See Dashboard Service license key configuration .

Data persistence requirements

Finally you will be prompted to choose whether you want to keep the data generated by the CAST Dashboards (logs etc.) within the container (**hit Enter**), or in a specific location on the server (enter a "root" folder that will be used for the CAST AIP data, for example **/var/opt/cast/dashboards/** (this path must NOT exist already)):

```
=====
Then, let us talk about data persistence:
```

```
All data are located inside the container, but you may want them to be persisted locally.
So that, you will be able to access them easily, to backup them, ...
You are going to set the storage location to persist your dashboards data.
Note that this folder will be created, it cannot already exist.
Note also that the data folder is located in /shared_data/dashboards inside the container.
=====
```

```
If you want to persist the folder where applications data are stored, give the absolute path to that folder.
if you do not want to persist those data, just "Enter" (without any value):
```

In a new installation, no existing containers will be detected, and the installer will then create a new Docker container with the default name **castdashboards_1**. You can enter a custom name if you prefer:

```
No old container to clean.
Container name could be: << castdashboards_1 >>, if you want a custom one, enter its name or Return if this one is ok for you:
```

The installer will then start the process of pulling the Docker image:

```
=====
Updating current image with latest one:
latest: Pulling from castaip/dashboards_linux
e7c96db7181b: Pull complete
f910a506b6cb: Pull complete
b6abafe80f63: Downloading [=====>] 31.06MB/54.94MB
391185a84357: Download complete
ae25fbce7c44: Downloading [=====>] 19.26MB/104MB
de2821164edb: Downloading [=====>] 19.38MB/90.54MB
cae7e1b50983: Waiting
2527d7890269: Waiting
586287663e34: Waiting
```

The container will then start:

```
=====
Starting a new container with ID:
Docker command: docker run -d -v /tmp/cast_dashboards_docker:/workspace --name castdashboards_1 -p 8080:8080 -p 8081:8081 -p 8082:8082 -p 8083:8083 castaip/dashboards_linux:latest
```

 The items are all downloaded directly from <https://hub.docker.com/>.

Check summary

When the image is up and running, a summary of the installation process will be displayed on screen:

Click to enlarge

```
SUCCESS: CAST dashboards container is ready to use
Health and Engineering dashboard: started successfully.
Health dashboard: started successfully.
Engineering dashboard: started successfully.
Security dashboard: started successfully.
SUCCESS: CAST dashboards container is ready to use
Container has started successfully.

=====
== Summary:
=====
Docker command:
docker run -d -v /tmp/cast_dashboards_docker:/workspace --name castdashboards_1 -p 8080:8080 -p 8081:8081 -p 8082:8082 -p 8083:8083 castaip/dashboards_linux:latest

Container name: castdashboards_1
See configuration file template here: /tmp/cast_dashboards_docker/dashboards.conf
Content is:
#
deployHE=yes
deployH=yes
deployE=yes
deployS=yes
cssHost=192.168.200.104
cssUser=operator
cssPassword=CastAIP
cssPort=2282

cssMeasureSchema=general_measure_125
cssCentralSchema=meudon_01_central
castDashboardLicense=
=====

Dashboards information:
=====
-- Health and Engineering dashboard url is:
-- http://localhost:8080
-- Port used for Health and Engineering Dashboard: 8080

-- Health dashboard url is:
-- http://localhost:8081/portal/index.html
-- Port used for Health Dashboard: 8081

-- Engineering dashboard url is:
-- http://localhost:8082/engineering/index.html
-- Port used for Engineering Dashboard: 8082

-- Security dashboard url is:
-- http://localhost:8083/security/index.html
-- Port used for Security Dashboard: 8083

== IMPORTANT:
Login information (case sensitive): admin | admin
This login information will be used for all dashboards.
=====
```

Check container and image

Run the following command to check that the image is up and running:

```
docker ps
```

```
root@ubuntudesktop:/home/james/AIP# docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS      PORTS                               NAMES
10a2d941b762  castaip/dashboards_linux:latest     "/bin/bash -c ./init."  30 minutes ago Up 29 minutes 0.0.0.0:8080-8083->8080-8083/tcp  castdashboards_1
```

Test Dashboard access

When the deployment is complete and depending on the installation choices you made, you should **test that you can access the deployed Dashboards, preferably from another server/workstation on the network**. The URLs and logins for each Dashboard are provided in the summary (shown above), but if you used the default settings, you can access the Dashboards as follows:

- http://<host_server>:8080
- http://<host_server>:8081/portal/index.html
- http://<host_server>:8082/engineering/index.html
- http://<host_server>:8083/security/index.html

Default configuration

All Dashboards are deployed with the following settings:

Authentication	Local authentication is enabled with one single user "admin" with the following credentials: <ul style="list-style-type: none">• admin/admin
-----------------------	--

Roles	The single user admin is granted the role ADMIN .
Authorization	The ADMIN user will have access to all applications across all dashboards.

Modifying the default configuration

If you would like to modify the default configuration for example, change the authentication method, configure roles, configure authorizations etc. the method you use depends on the installation choices you have made. See below. You can refer to the following documentation for more information about the options available:

- [User authentication](#)
- [First login and become admin](#)
- [Dashboard Service license key configuration](#)
- [User roles](#)
 - [User roles - 2.x and above](#)
 - [User roles - 1.x](#)
- [Data authorization](#)
 - [Data authorization - 2.x and above](#)
 - [Data authorization - 1.x](#)

 If you make any changes, restart the container to ensure the changes are taken into account.

```
docker restart <my_container_id>
```

Simple mode choice

In **simple mode**, all configuration files are located within the container, therefore you will need to access them as follows where **<container_name>** is the name of the container created by the installer in Docker - the default is **castdashboards_1**:

```
docker exec -it <container_name> /bin/bash
```

For example:

```
docker exec -it castdashboards_1 /bin/bash
```

You will find the configuration files for each dashboard in the **/cast** sub-folder:

```
bash-4.4# ls
bin          etc          lib          opt          run          srv          usr
cast        home        media        proc         sbin        sys          var
dev         init_container.sh  mnt         root        shared_data tmp          workspace
```

Then each dashboard is located in its own folder:

- **engineering**
- **health**
- **health_engineering**
- **security**

```
bash-4.4# cd cast
bash-4.4# ls
engineering  health  health_engineering  security
```

Use your favourite editor to configure the file you need:

```
bash-4.4# cd cast
bash-4.4# ls
engineering      health           health_engineering  security
bash-4.4# cd engineering
bash-4.4# ls
amd64            configurations   dashboard-service.exe  dashboard-servicew.exe  startingDashboard.log
app.pid          cast-engineering-dashboard-2.0.1-31.jar  dashboard-service-uninstall.bat  shutdown.sh             startup.bat
bash-4.4# cd configurations
bash-4.4# ls
application.properties  config          indexesDirectory  license.xml  logs          monitoring          security
authorizations.xml      domains.properties  license.key        log4j2-spring.xml  metrics-documentation  roles.xml          users.properties
bash-4.4# nano roles.xml
```

Finally, to exit the container and return to your local console, just enter **exit**. If you have made any changes, restart the container.

Advanced mode choice with data stored outside the container

If you chose to store the data outside the container in **advanced mode**, you can edit the file with your favourite editor, save and then restart the container if you have made any changes.