# Einrichtung der FIWARE-Infrastruktur mit Docker

## Inhaltsverzeichnis

[Inhaltsverzeichnis 1](#_Toc518485960)

[Benötigte Software 2](#_Toc518485961)

[Initialisierung der Docker-Container 2](#_Toc518485962)

[Initialisierung von Wirecloud 3](#_Toc518485963)

[Einrichtung der Security-Komponenten 4](#_Toc518485964)

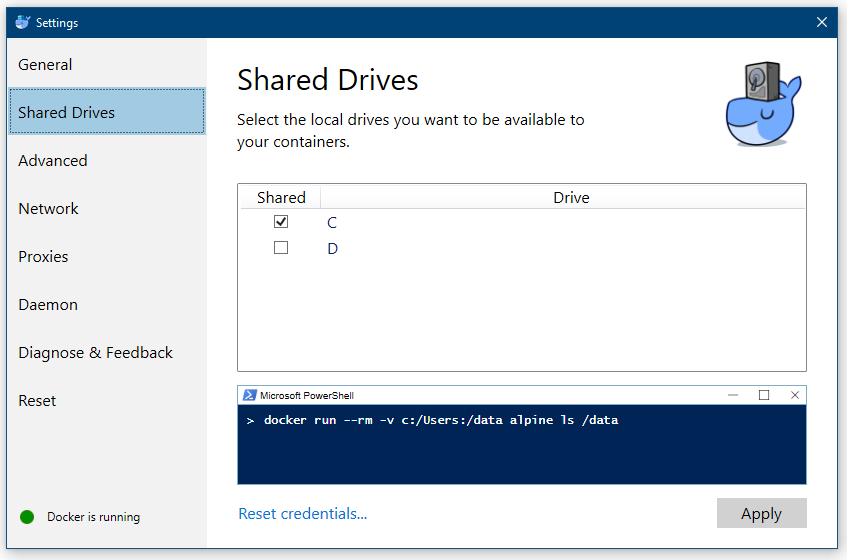
[Einrichtung des Dashboards 10](#_Toc518485965)

[Erstellen von Sensorwerten 12](#_Toc518485966)

## Benötigte Software

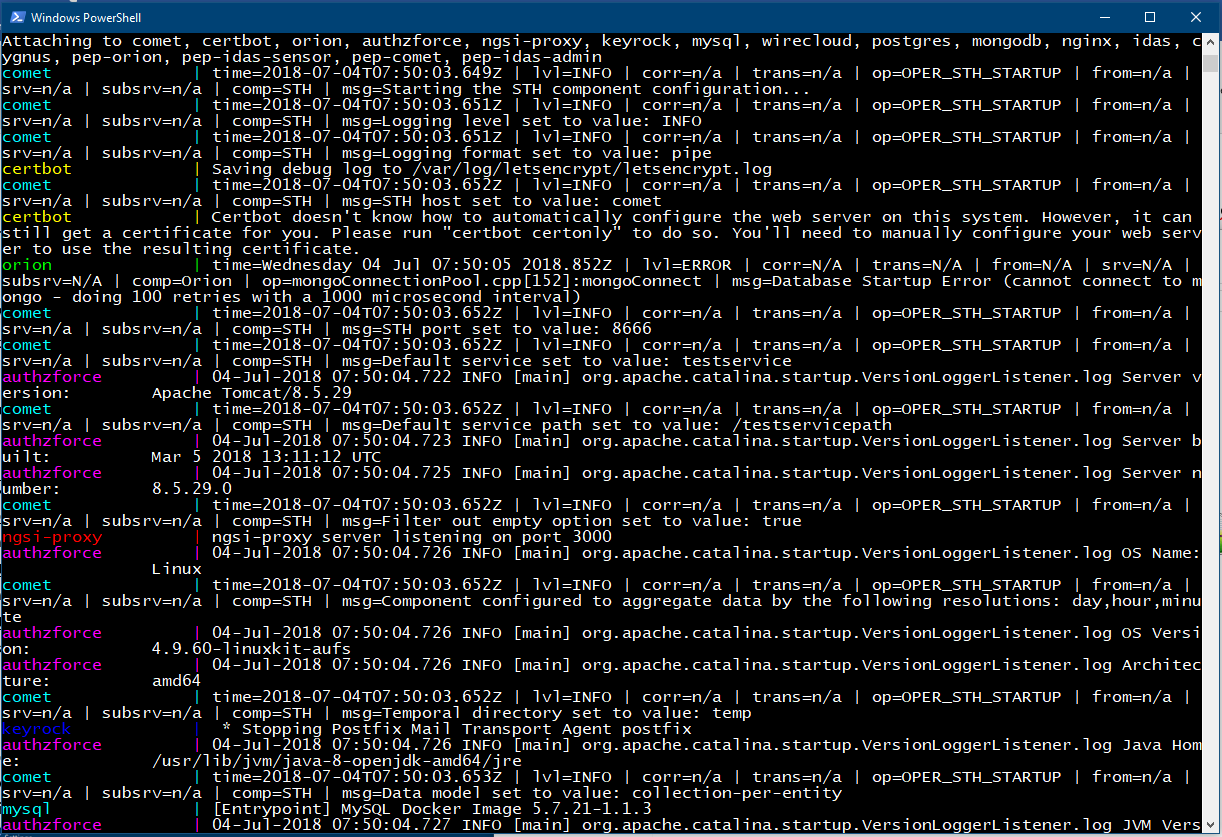
* Docker Community Edition (inklusive Docker Compose)  
  Link: <https://docs.docker.com/docker-for-windows/install/>
* Python 2.7  
  Link: <https://www.python.org/download/releases/2.7/>
* Python-Bibliothek „requests“  
  Installation: Befehl ***pip install requests*** in Konsole ausführen

## Initialisierung der Docker-Container

Nach dem Starten des Docker-Clients muss als Erstes die Laufwerksfreigabe im Einstellungsmenü festgelegt werden:

Dieser Schritt ist notwendig, da es sonst zu Problemen mit dem Einhängen von Dateien und Volumes in Containern kommen kann.

Danach muss die Datei „docker-compose.yml“ im Ordner „[…]DockerInfrastructure\Docker“ in „docker-compose.yml.security“ (oder ähnlich) umbenannt werden. Die Datei „docker-compose.yml.no\_security“, welche im nächsten Schritt benötigt wird, muss hingegen in „docker-compose.yml“ umbenannt werden.

Als nächstes muss eine PowerShell-Konsole im Ordner „[…]DockerInfrastructure\Docker“ geöffnet werden. Die Initialisierung der Container wird mit dem Befehl ***docker-compose up*** gestartet.  
Docker lädt nun die benötigten Images von Docker Hub und bereitet die Container für den ersten Start vor. Wenn die Initialisierung abgeschlossen ist, werden die Log-Meldungen der einzelnen Container angezeigt:

## Initialisierung von Wirecloud

Bevor Wirecloud verwendet werden kann, muss erst ein Benutzer mit Administratorrechten angelegt werden. Dafür muss der folgende Befehl in einem weiteren PowerShell-Fenster ausgeführt werden::

***docker exec -it wirecloud python ./manage.py createsuperuser***

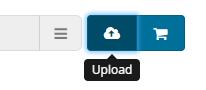
## Einrichtung des Dashboards

Nachdem nun die Basiskomponenten der FIWARE-Infrastruktur fertig konfiguriert sind, muss das Dashboard für die Verwaltung der Datenbankobjekte und die Darstellung der Sensorwerte eingerichtet werden. Dazu muss als Erstes das Wirecloud-Portal unter der Adresse [**https://localhost**](https://localhost) im Browser geöffnet werden. Nach einem Klick auf den Link **Sign in** in der rechten oberen Ecke kann der zuvor angelegte Benutzer eingeloggt werden.

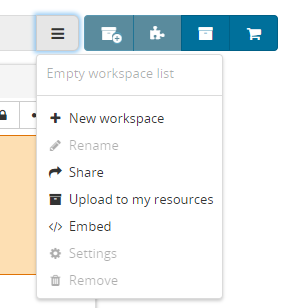
Nach der erfolgreichen Anmeldung wird der Workspace-Browser des jeweiligen Benutzers angezeigt. Über einen Klick auf den Button **My Resources** (siehe nachfolgende Grafik) gelangt man nun in die Übersicht der eigenen Ressourcen, wo die benötigten Widgets und Operatoren für das Dashboards hochgeladen werden können.

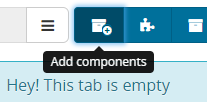


Durch einen Klick auf Upload (siehe nächste Abbildung) wird das Upload-Fenster geöffnet, von dem aus Ressourcen vom Host-Rechner hochgeladen werden können.



Die benötigten Dateien für das Dashboard befinden sich im Ordner „[…]DockerInfrastructure\Mashup“.

Nachdem der Upload der fünf Dateien erfolgreich abgeschlossen wurde, muss im Workspace-Browser über **New workspace** (siehe nachfolgende Grafik) ein neuer Workspace erstellt werden. Die Benennung des Workspace ist frei wählbar.

In der Basis-Ansicht des neu erstellten Workspace kann nun durch einen Klick auf **Add components** (zu sehen in nachfolgender Abbildung) das hochgeladene Mashup importiert werden.

Unter dem Reiter „Mashups“ muss nun das Mashup „Dashboard“ über das Plus-Symbol hinzugefügt werden. War der Vorgang erfolgreich, sollten am unteren Rand der Workspace-Ansicht neue Tabs dazugekommen sein („Karte“, „Diagramme“, etc.). Durch einen Klick auf dem jeweiligen Tab kann die entsprechende Dashboard-Ansicht angezeigt und mit den Widgets interagiert werden.

Für das Anlegen der benötigten Datenbankobjekte muss als nächstes auf den Tab „Batch Add“ gewechselt werden. Mit dem dort vorhandenen Widget können mehrere Datenbankobjekte gleichzeitig per JSON-Eingabe erstellt werden. Dafür muss nur die jeweilige JSON-Struktur in das leere Textfeld des Widgets kopiert und danach auf den Button **Send** geklickt werden. Die Dateien mit den Objekten im JSON-Format sind im Ordner „[…]DockerInfrastructure\DBObjectsJSON“ zu finden.  
Wichtig: Die Datenbankobjekte müssen immer in der Reihenfolge **Services – Entities – Devices – Subscriptions** erstellt werden. Wird diese Reihenfolge nicht eingehalten, werden unter Umständen fehlende Objekte intern automatisch generiert, wodurch es bei der Objekterstellung mit dem Batch-Add-Widget zu Komplikationen kommen kann.

Nach der Erstellung der Datenbankobjekte kann über die Widgets im Tab „Admin Tools“ überprüft werden, ob alle Objekte ordnungsgemäß erstellt wurden. Auch das Ändern und Löschen von Objekten sowie das Anlegen neuer Datenbankobjekte kann über diese Widgets durchgeführt werden.

## Erstellen von Sensorwerten

Um auch ohne echte Sensoren die FIWARE-Infrastruktur mit Sensorwerten versorgen zu können, werden künstliche Sensorwerte mithilfe eines Sensor-Mock-Scripts an den Server übertragen.

Dafür kann ein PowerShell-Fenster im Ordner „[…]DockerInfrastructure\SensorMockScript“ geöffnet werden, aus dem mit dem folgenden Befehl Messwerte vom Sensor „Tram\_1“ an den Server übermittelt werden:

***python .\SendCombinedValues.py Dev\_Tram\_1 apimobile .\tram\_1-values.txt***

Das Skript sendet nun alle fünf Sekunden einen (in der Datei „tram\_1-values.txt“ hinterlegten) Messwert an den Server. Um das Skript zu stoppen, muss nur die Tastenkombination **Strg+C** getätigt werden.

Die Befehle für die restlichen Sensoren sind in der Datei „Commands\_no\_security.txt“ im selben Ordner zu finden.