Kubernetes with Containers and DevOps Workshop

Hands-on lab step-by-step

Aralık 2018

1. [Run app locally to test components]

The lab files must be cloned to the local machine to complete all of the exercises for the day.

* Open a terminal on the jumpbox to bring up a command line
* Clone the Github repo via the command line

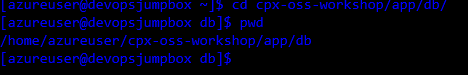
git clone https://github.com/eozkurt/cpx-oss-workshop

**Database layer - MongoDB**

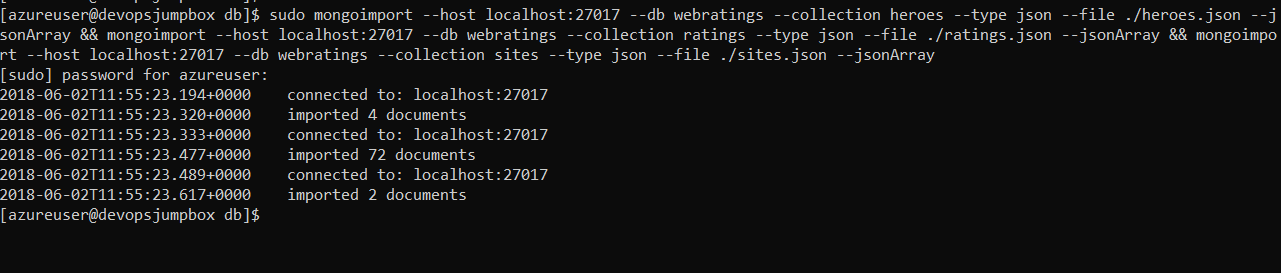
The underlying data store for the app is [MongoDB](https://www.mongodb.com/). It is already running. We need to import the data for our application.

* Import the data via the command line

cd ~/cpx-oss-workshop/app/db/



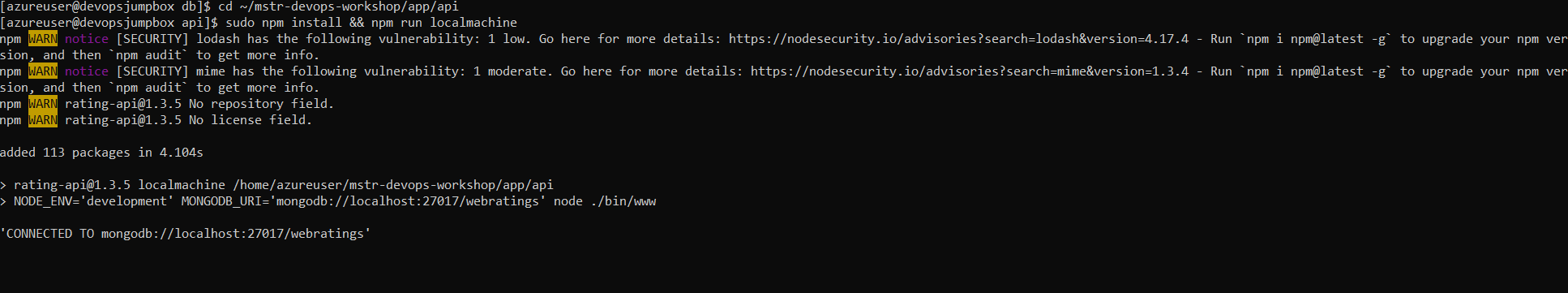
sudo mongoimport --host localhost:27017 --db webratings --collection heroes --type json --file ./heroes.json --jsonArray && mongoimport --host localhost:27017 --db webratings --collection ratings --type json --file ./ratings.json --jsonArray && mongoimport --host localhost:27017 --db webratings --collection sites --type json --file ./sites.json --jsonArray



**API Application layer - Node.js**

The API for the app is written in javascript, running on [Node.js](https://nodejs.org/en/) and [Express](http://expressjs.com/)

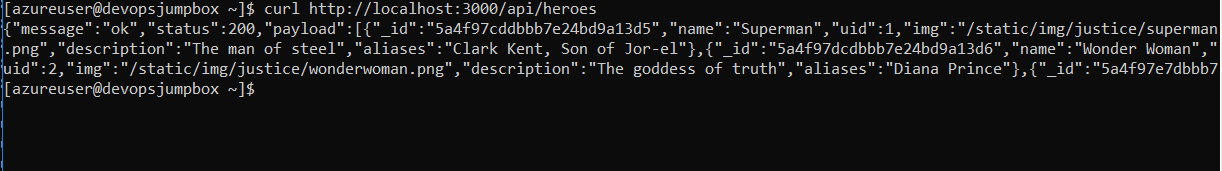
* Update dependencies and run app via node in the command line
  + 1. cd ~/cpx-oss-workshop/app/api
    2. npm install && npm run localmachine



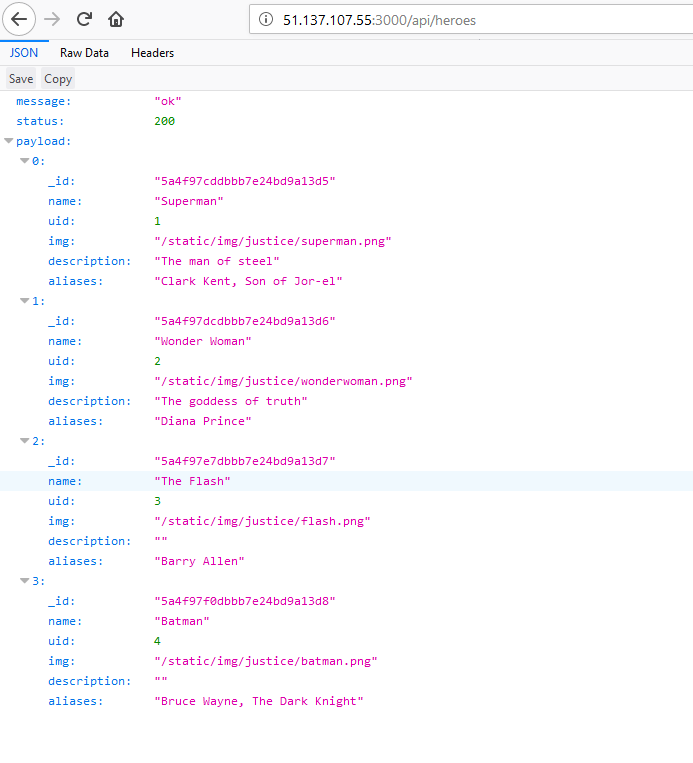
**Open a new terminal session** on the **jumpbox** and test the API (Do not forget to change {PUBLIC\_IPADDRESS\_VM} with your jumpbox vm’s ip address)

using curl in the command line

* + 1. ssh azureuser@{PUBLIC\_IPADDRESS\_VM}
    2. curl <http://localhost:3000/api/heroes>



or use the browser on your local computer and navigate to [http:// {PUBLIC\_IPADDRESS\_VM}:3000/api/heroes](http://localhost:3000/api/heroes)



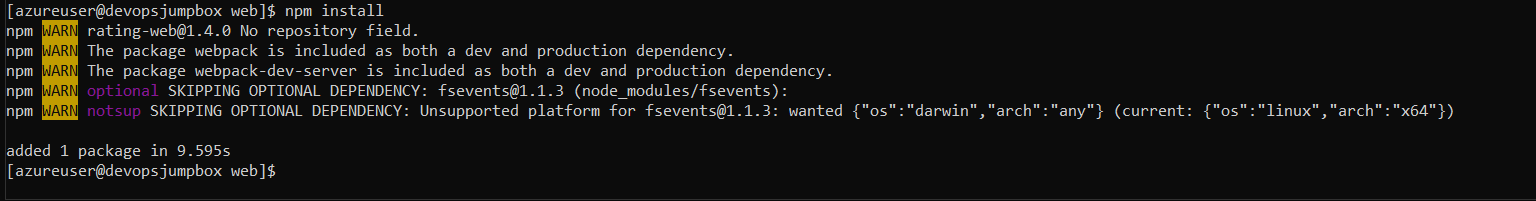
**Web Application layer - Vue.js, Node.js**

The web frontend for the app is written in [Vue.js](https://vuejs.org/Vue" \o "Vue.js Homepage), running on [Node.js](https://nodejs.org/en/) with [Webpack](https://webpack.js.org/)

* Continue new Terminal session
* Update dependencies and run app via node

cd ~/cpx-oss-workshop/app/web/

npm install && npm run localmachine



* Test web locally

use the browser on the jumpbox and navigate to http://{PUBLIC\_IPADDRESS\_VM}:8080

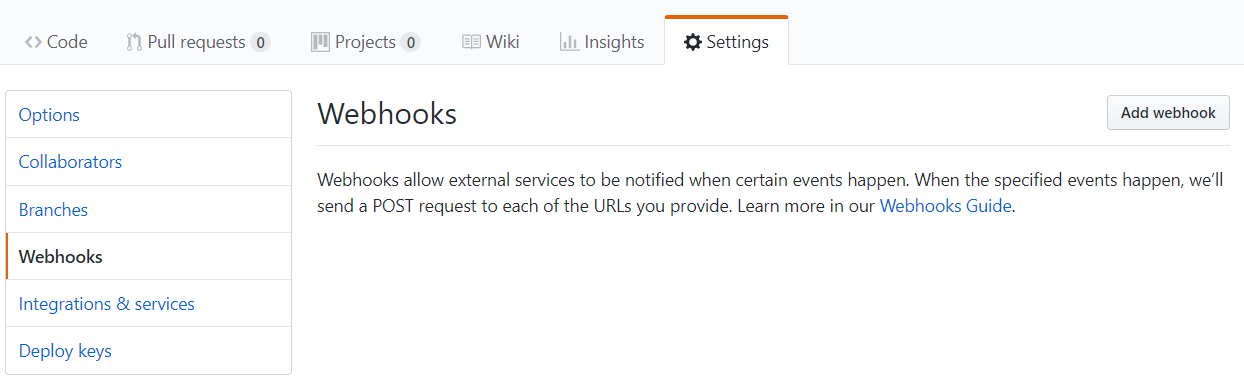
**Clean-up**

* **Close the web and api apps in the terminal windows by hitting ctrl-c in each of the corresponding terminal windows**

**Create GitHub webhook**

To configure the integration with GitHub, open the <https://github.com/eozkurt/cpx-oss-workshop> from sample repo. To fork the repo to your own GitHub account, select the Fork button in the top right-hand corner.

Once you fork the repo follow these steps to create web hook.

* Go to the settings page of your repository or organization. From there, click **Webhooks**, then **Add webhook**.
* Typehttp://{PUBLIC\_IPADDRESS\_VM}:8081/github-webhook/ into the Payload URL field. DO NOT forget to change PUBLIC\_IPADDRESS\_VM with your jumpbox public ip address.
* Choose **application/json** from the Content Type field.
* Leave blank the Secret field.
* Click **Send me everything** to trigger all events.
* Once you finished, be sure that you have clicked **Active** to enable this webhook and Click **Add webhook**.