

FIWARE - How to begin with Orion Context Broker

1. Download the Orion VirtualBox image: <http://bit.ly/fiware-orion024-vbox>
2. Once you have the Orion VirtualBox image, configure it on your VirtualBox and run. 1GB RAM and 8GB HD storage were enough.
3. For Network Settings, you can choose NAT. You have to setup the following "Port forwarding" rule in VirtualBox:
 - a. Protocol - TCP
 - b. HOST IP - Your host IP
 - c. HOST Port - 1026 (default)
 - d. Guest IP - Your VM IP
 - e. Guest Port - 1026
4. Now, you have a CentOS with Orion installed. Login and user for CentOS are both: fiware.
5. Test if you can ping your host machine ip and if you have access to Internet.
6. Update the Orion version with the command: `sudo yum install contextBroker`
7. To test, you can use the following command from your host machine: `curl http:your_host_ip:1026/version` (you will see the version of Orion)

FIWARE - IdM Keyrock

1. The easier way to IdM Keyrock running is using docker. With docker installed in your machine, do: `sudo docker run -d --name idm -p 8000:8000 -p 5000:5000 -t fiware/idm`
2. This command will configure a container with IdM Keyrock on your machine, with user and password set to 'idm', Horizon mapped to port 8000 and Keystone mapped to port 5000.
3. To start the container, do: `sudo docker start idm`
4. To access the container, through bash: `sudo docker exec -it idm bash`
5. To test, you can use the following command from your host machine: `curl http://your_host_ip:5000` or `http://your_host_ip:8000` (you must see the json message about openstack keystone versions or the html code related to horizon start page)
6. Access horizon at localhost:8000 and click on Applications ---> Register.
7. Fill all the inputs. The application url is the url Keyrock accepts OAuth requests. The callback url is the url that the browser redirects to, after the OAuth flow is finished.

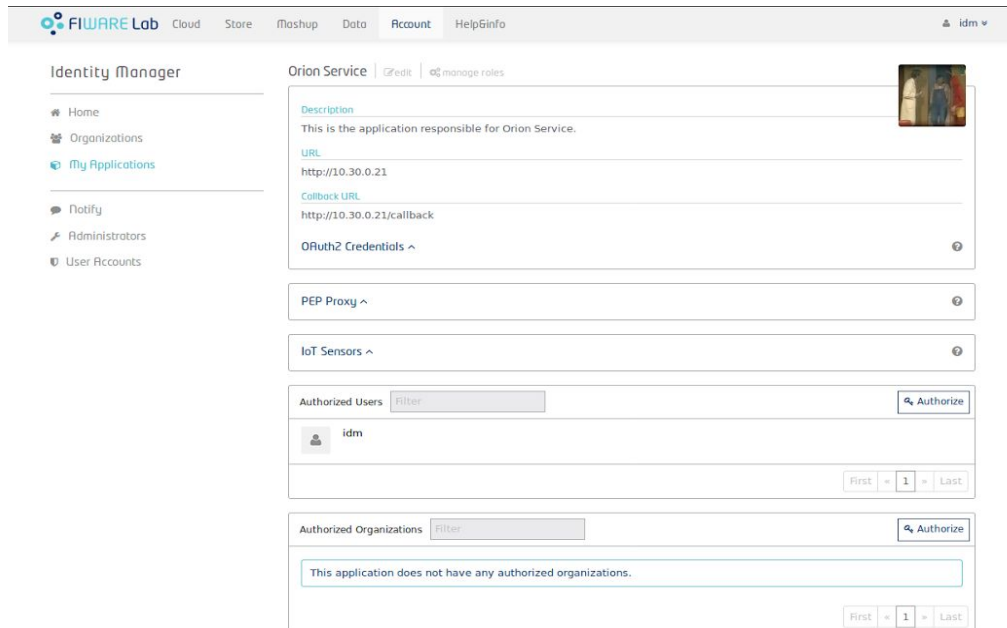
FIWARE - PEP Proxy Wilma

1. With nodejs and npm installed in your machine, clone this repository: `git clone https://github.com/ging/fiware-pep-proxy.git`
2. Enter in the fiware-pep-proxy folder and run: `npm install`

- After the installation process, copy the content of config.template to config.js file (create it). Configure the params config.app_host and config.app_port with your host address and port.
- Run the server: nodejs server

FIWARE - How to integrate Orion, Wilma and Keyrock

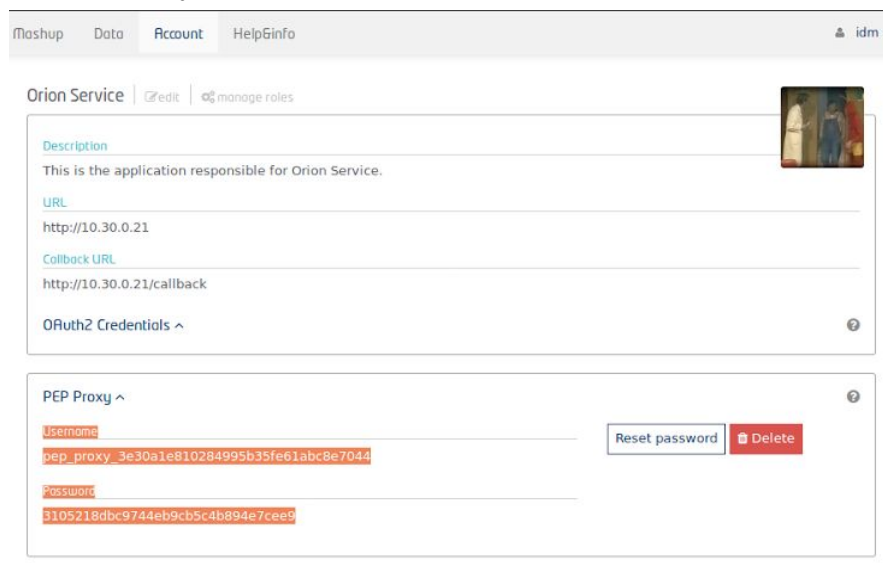
- Once you have Orion, Wilma and Keyrock running and tested, let's configure the integration between all of them
- With Keyrock running (http://localhost:8000), access your application configured



The screenshot shows the FIWARE Lab Identity Manager interface. The top navigation bar includes links for Cloud, Store, Mashup, Data, Account, and Help/Info. The left sidebar lists navigation options: Home, Organizations, My Applications, Notify, Administrators, and User Accounts. The main content area displays the configuration for the 'Orion Service'. It includes fields for Description, URL (http://10.30.0.21), and Callback URL (http://10.30.0.21/callback). Below these are expandable sections for OAuth2 Credentials, PEP Proxy, and IoT Sensors. The 'Authorized Users' section shows a list with one user, 'idm'. The 'Authorized Organizations' section shows a message: 'This application does not have any authorized organizations.'

2015 © FIWARE. The use of FIWARE Lab services is subject to the acceptance of the [Terms and Conditions](#), [Privacy Policy](#) and [Cookies Policy](#).

- Get the PEP Proxy username and password



The screenshot shows the FIWARE Lab Identity Manager interface, specifically the 'PEP Proxy' configuration section. It displays fields for Username and Password. The Username field contains the value 'pep_proxy_3e30a1e810284995b35fe61abc8e7044' and the Password field contains '3105218dbc9744eb9cb5c4b894e7cee9'. There are 'Reset password' and 'Delete' buttons next to the fields. The top navigation bar and sidebar are also visible.

4. Edit your Wilma PEP Proxy config.js file with these information:
config.account_host = 'http://idm_ip:8000'; //ip from IdM Keyrock instance

config.keystone_host = 'idm_ip'; //ip from IdM Keyrock instance
config.keystone_port = 5000;

config.app_host = 'orion_ip'; //ip from Orion instance
config.app_port = '1026';
// Use true if the app server listens in https
config.app_ssl = false;

// Credentials obtained when registering PEP Proxy in Account Portal
config.username = 'pep_proxy_3e30a1e810284995b35fe61abc8e7044';
config.password = '4c87c6db2a9a43a4a9d9e1e95759df62';

5. Now, you have to request a valid token in the IdM Keyrock. To get this, do a post request with the following information:

POST to "http://idm_ip:8000/oauth2/token"

Payload:

grant_type=password&username=YOUR_USERNAME&password=YOUR_PASSWORD&client_id=YOUR_CLIENT_ID&client_secret=YOUR_CLIENT_SECRET

In the payload, change all "YOUR_" params to your own information: username and password from a valid registered user in the Keyrock and client_id and client_secret from a valid registered application in the Keyrock.

Headers:

'Content-Type': 'application/x-www-form-urlencoded', 'Authorization': 'Basic AUTH_HEADER'
Where AUTH_HEADER must be changed to a Base64 encoded of this information:
"client_id :client_secret" - something like base64(client_id + ":" + client_secret)

6. After get a valid token from IdM Keyrock, you can make some request for Orion, with the following information:

GET or POST to http://idm_ip/v2/entities (note the idm_ip because we have the Wilma PEP Proxy running on the IdM container and protecting our Orion instance. So, all the requests to Orion will pass by Wilma to be authenticated and authorized)

Headers:

'Content-Type': 'application/json', 'Accept': 'application/json', 'X-Auth-Token': 'your valid IdM token obtained in the last step'

Payload:

According to Orion operations. See the NGSI v2 to learn about the JSON formats.