Instruction

Introduction

micROS-cloud server

Environment

- Ubuntu 14.04.3 LTS
- Python 2.7.6
- Docker version 1.11.2

Installation

Docker installation

(1) Follow the tutorial to get Docker installed: https://docs.docker.com/engine/installation/linux/ubuntulinux/
(2) Install and Create a Docker Swarm: https://docs.docker.com/swarm/install-w-machine/

Flask installation

(1) Log into your machine as a user with sudo or root privileges.
(2) Open a terminal window.
(3) Update package information
(4) Install python-pip:
```
$ sudo apt-get update
$ sudo apt-get install python-pip
```
(5) Install flask:
```
$ sudo pip install flask
```
(6) Install database:
```
$ sudo pip install flask_sqlalchemy
```
(7) Install flask_login
```
$ sudo pip install flask_login
```
(8) Install flask-wtf

```
$ sudo pip install flask-wtf
```

Docker-py installation

```
$ sudo pip install docker-py
```

micROS-cloud server start

(1) Download and unzip the micROS-cloud package.

(2) Prepare the base image:

1) Find the dockerfile in the folder: (the path of micROS-cloud-master)/micROS-cloud-master/base-image and build it.

```
$ cd ../micROS-cloud-master/base-image
$ sudo docker build .
```

2) Get the id of the new created image:

```
$ docker images
```

3) Rename the image:

```
$ sudo docker tag <id> ros:my
```

(3) Start the micROS- cloud server with the command:

```
$ cd ../micROS-cloud-master
$ python run.py
```

Note: If there is an error: Unable to list the containers. Reason: ('Connection aborted.', error(13, 'Permission denied')). Please run the command with the system permissions.

Function introduction

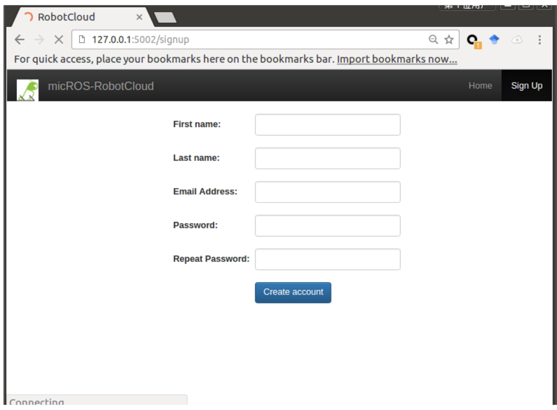
Open the browser and enter the IP address.

```
<server ip address>:5002
```

For example: localhost:5002

Registration and login

Click the Sign Up button to create an account and then login:


RobotCloud
127.0.0.1:5002/signup
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
micROS-RobotCloud
Home
Sign Up
Create account
Connecting

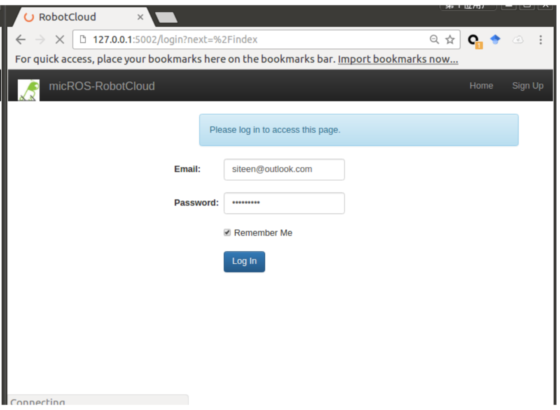
First name:

Last name:

Email Address:

Password:

Repeat Password:


RobotCloud
127.0.0.1:5002/login?next=%2Findex
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
micROS-RobotCloud
Home
Sign Up
Log In
Connecting

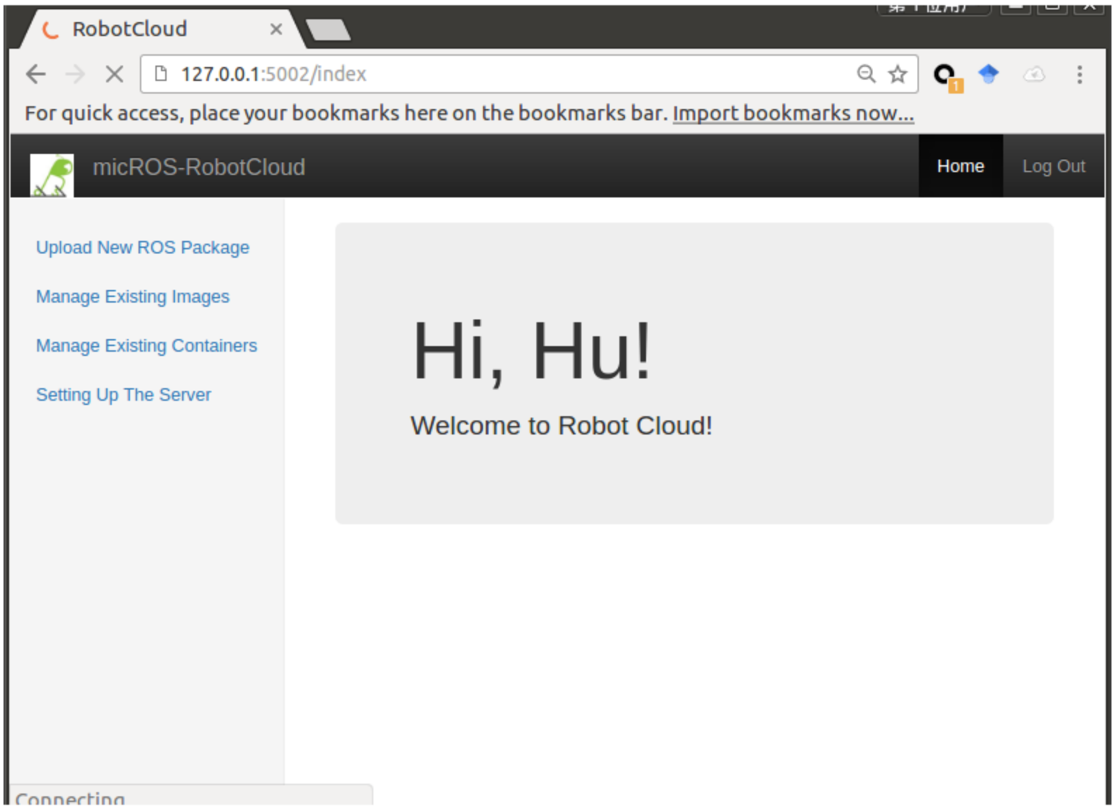
Please log in to access this page.

Email: siteen@outlook.com

Password: ••••••••

☑ Remember Me

After login, you will see the welcome page:



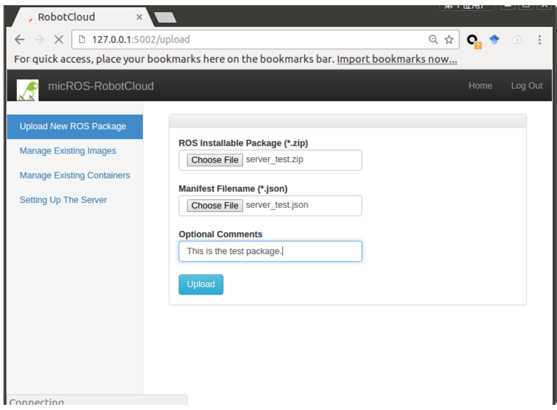
Hi, Hu!

Welcome to Robot Cloud!

Upload ROS package

In the *Upload New ROS Package* page, you can upload a ROS package and a json file to define a cloud service. After uploading successfully, you can download a proxy for you robot to access the service.

For example, you can use the ROS package and json file in the *test-case* folder.


RobotCloud

127.0.0.1:5002/upload

For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

micROS-RobotCloud

Home

Log Out

Upload New ROS Package

Manage Existing Images

Manage Existing Containers

Setting Up The Server

ROS Installable Package (*.zip)

Choose File server_test.zip

Manifest Filename (*.json)

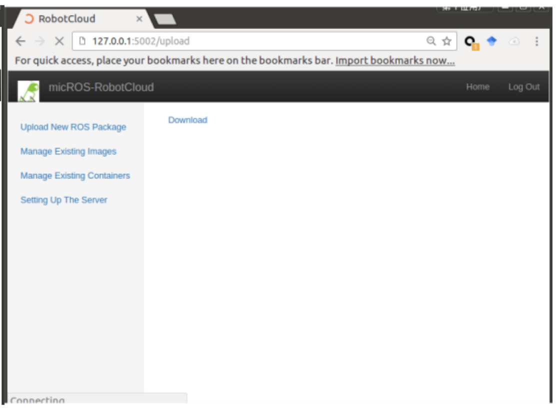
Choose File server_test.json

Optional Comments

This is the test package.

Upload

Connecting


RobotCloud

127.0.0.1:5002/upload

For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

micROS-RobotCloud

Home

Log Out

Upload New ROS Package

Manage Existing Images

Manage Existing Containers

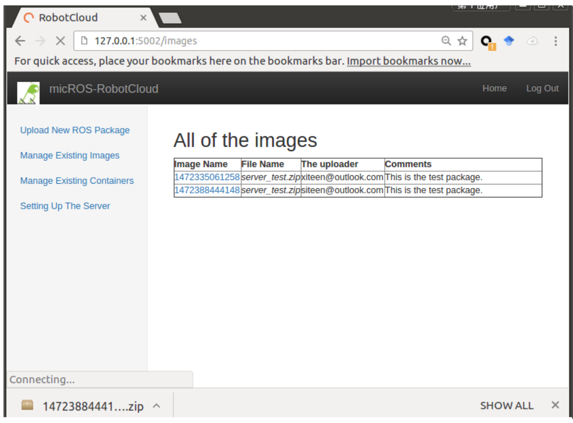
Setting Up The Server

Download

Connecting

Manage the existing images

In the *Manage Existing Images* page, you can get the information of every specific docker image.


RobotCloud

127.0.0.1:5002/images

For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

micROS-RobotCloud

Home

Log Out

Upload New ROS Package

Manage Existing Images

Manage Existing Containers

Setting Up The Server

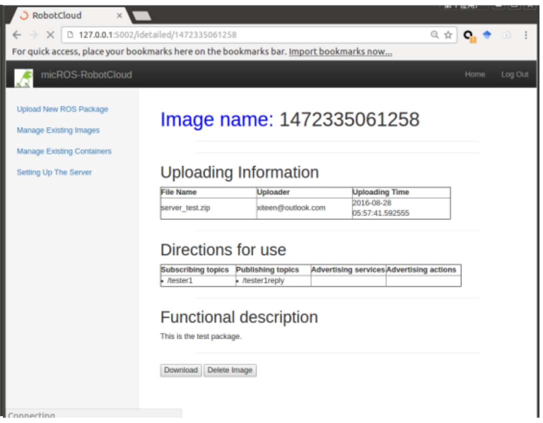
All of the images

Image Name	File Name	The uploader	Comments
1472335061258	*server_test.zip*	xiteen@outlook.com	This is the test package.
1472388444148	*server_test.zip*	siteen@outlook.com	This is the test package.

Connecting...

14723884441....zip

SHOW ALL


RobotCloud

127.0.0.1:5002/idetailed/1472335061258

For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

micROS-RobotCloud

Home

Log Out

Upload New ROS Package

Manage Existing Images

Manage Existing Containers

Setting Up The Server

Image name: 1472335061258

Uploading Information

File Name	Uploader	Uploading Time
server_test.zip	xiteen@outlook.com	2016-08-28 05:57:41.592555

Directions for use

Subscribing topics	Publishing topics	Advertising services	Advertising actions
• /tester1	• /tester1reply		

Functional description

This is the test package.

Download

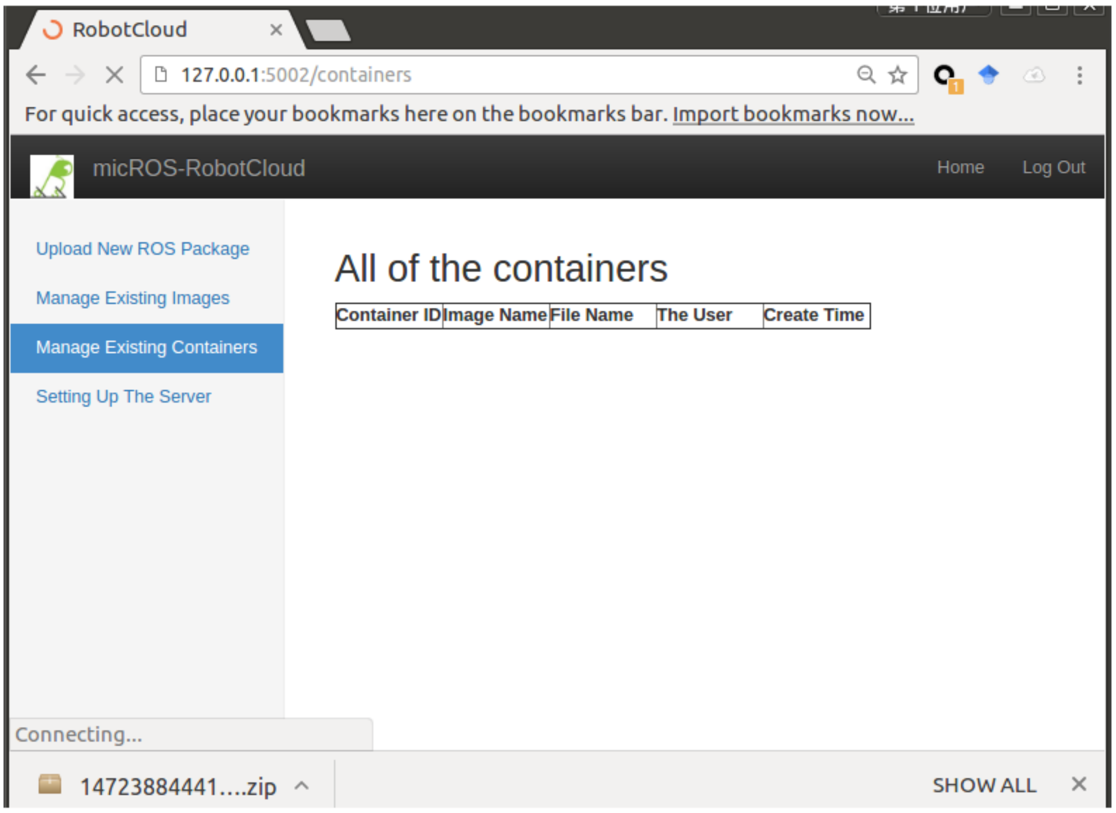
Delete Image

Connecting

Click the image name and you can get the details of the image. In this page you can also download the proxy or delete the image.

Manage the existing containers

In the *Manage Existing Containers* page, you can get the information of containers and you can stop/start a container or remove a container.

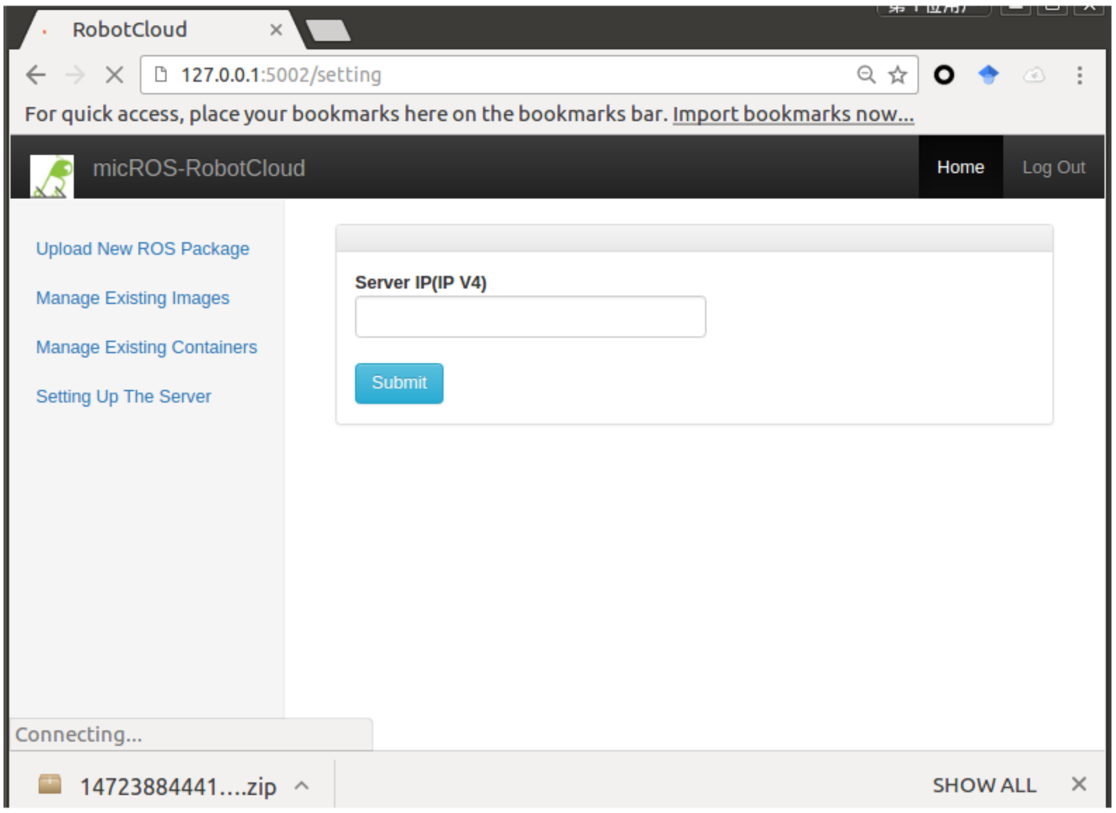


All of the containers

Container ID	Image Name	File Name	The User	Create Time

Setting

In the *Setting* page, you can set the server ip address for a remote robot to connect to the service.



micROS-cloud client

Environment

- Python 2.7.6
- ROS indigo

Installation

ROS installation

The Robot Operating System (ROS) is a flexible framework for writing robot software. It is a collection of tools, libraries, and conventions that aim to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platforms.

(1) Follow the tutorial to install the ROS: http://wiki.ros.org/indigo/Installation/Ubuntu

(2) Create a ROS Workspace:
http://wiki.ros.org/ROS/Tutorials/InstallingandConfiguringROSEnvironment

micROS-cloud client start

After upload the ROS package you can get the proxy package.

(1) Download and unzip the proxy package.

(2) Copy the package to the ~/catkin_ws/src

(3) Open the terminal and source the *setup.bash* file

```
$ cd catkin_ws/src/client/cloudproxy
$ source setup.bash
```

Note: If there is an error: bash: ~/catkin_ws/src/client/cloudproxy/_setup_util.py: Permission denied
Failed to run '"~/catkin_ws/src/client/cloudproxy/_setup_util.py" ': return code 126.
Please run the following command:

```
$ chmod +x _setup_util.py
```

(4) Run the proxy:

```
$ roslaunch cloudproxy client.launch
```

Note: If there is an error: ERROR: cannot launch node of type [cloudproxy/client]: can't locate node [client] in package [cloudproxy]. Please run the following command:

```
$ cd ~/catkin_ws/src/client/cloudproxy/lib/cloudproxy
$ chmod +x client
$ cd ../..
$ source setup.bash
$ roslaunch cloudproxy client.launch
```

(5) After the proxy is running, you can run the other ROS node to publish the topics that the remote running server required.