# Aerohive Selenium Basic Introduce

Due to this lib is based on remote webdriver(Selenium), you can use a client to remote control many servers which has been enabled selenium services already to simulate GUI execution. If you want to start use it, please follow the steps as below.

## Init Environment:

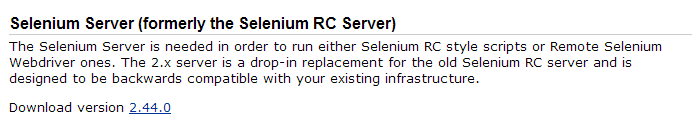
You can set up your Topo such as below



Pic1 Example Topo

1. Prepare for client: Install python and python-selenium(you can use pip to find it);
2. Prepare for servers: Install JRE/JDK and download remote server file from the link as below;

<http://www.seleniumhq.org/download/>



1. Checkout the folder “AerohiveSelenium” to your client, and add the path to python path;
2. Use command “ java -jar /${path}/selenium-server-standalone-2.x.x.jar” to enable selenium service on every servers;

## Structure introduce:

Folder page: We collect test pages’ html elements under this folder;

Folder profile: We collect the browser profile you need to init browser under this folder;

Folder scripts: We write your test produce under this folder;

## Start your first project

The first project I will demo how to open browser, get to baidu, search “hello word”.

First of all, you should have xpath basic knowledge, if you are ready, let’s start.

### Step1: Elements part

Create file page/example/\_\_init\_\_.py, we put all element locators in this file.

# -\*- coding: UTF-8 -\*-

from selenium.webdriver.common.by import By

baidu\_keyword\_input = (By.XPATH, *'//input[@id="kw"]'*)

baidu\_search\_btn = (By.XPATH, *'//input[@id="su"]'*)

### Step2: Operation part

#### Create file scripts/example/\_\_init\_\_.py, we write operation func in this file.

# -\*- coding: UTF-8 -\*-

from AerohiveSelenium import \*

from AerohiveSelenium.page.example import \*

import time, re

class **example**(object):

def **\_\_init\_\_**(*self*):

*self*.w = WebGUI()

def **search**(*self*, url, keyword):

*self*.w.driver.get(url)

*self*.w.wait\_until\_element\_displayed(baidu\_keyword\_input)

*self*.w.info(*'Get url successfully'*, True)

*self*.w.input(baidu\_keyword\_input,keyword)

*self*.w.click(baidu\_search\_btn)

*self*.w.info(*'Search keyword successfully'*, True)

#### Create file scripts/example/search.json, we write default value in this file.

{

"visit": {"url":"http://www.baidu.com/",

"keyword":"hello world"}

}

#### Create file scripts/example/search.py, we write operation logic in this file.

# -\*- coding: UTF-8 -\*-

from AerohiveSelenium.scripts.example import example

def **key\_search**():

con = example()

con.search(con.w.get\_value(*"visit.url"*), con.w.get\_value(*"visit.keyword"*))

if \_\_name\_\_ == *'\_\_main\_\_'*:

key\_search()

#### Debug your first project

My folder path is ‘/home/python/AerohiveSelenium/AerohiveSelenium’, we should add python path firstly.

*export PYTHONPATH=$PYTHONPATH:/home/python/AerohiveSelenium/*

Server 1’s IP is 10.155.81.221, so we can use the command as below to start your Aerohive Selenium travel.

*root@Linux227:~# python /opt/project/svn/AerohiveSelenium/AerohiveSelenium/scripts/example/search.py -r http://10.155.81.221:4444/wd/hub --parameters visit.keyword=hello -t ff -l info -f /tmp/test.log*

*2014-11-07 15:34:00,729 - AerohiveSeleniumTest - INFO - \_\_init\_\_.py:302 - dump of webui*

*remote-selenium = http://10.155.81.221:4444/wd/hub*

*browser-type = ff*

*browser-profile = /opt/project/svn/AerohiveSelenium/AerohiveSelenium/profile/ff*

*log-level = info*

*log-file = /tmp/test.log.html*

*log-pic-dir = /tmp/test.log\_pic*

*para\_dict = {u'visit.url': u'http://www.baidu.com/', u'visit.keyword': 'hello'}*

*preserve-session = False*

*session-id = None*

*2014-11-07 15:34:08,037 - AerohiveSeleniumTest - INFO - \_\_init\_\_.py:302 - Get url successfully pic:./test.log\_pic/000\_.png*

*2014-11-07 15:34:10,107 - AerohiveSeleniumTest - INFO - \_\_init\_\_.py:302 - Search keyword successfully pic:./test.log\_pic/001\_.png*

*root@Linux227:~#*

You can get log file and pic as below…your first program is successful now. Have fun!

