SmartAT instruction

This document provides detailed instructions on how to deploy SmartAT and start it, please follow the steps one by one and make sure every last step get a positive response before moving on.

**Deploy SmartAT**

**Requirement list,**

1, PC with Ubuntu 12.04 LTS and later release.

2, Functional python 2.7.X environment.

3, Download the source code of SmartAT from Github.

**Step 1,**

Install required runtime libraries.

Execute following commands in terminal.

*sudo apt-get install libssl-dev python-dev libevent-dev*

**Step 2,**

Install pip in order to install more python libraries.

Execute following commands in terminal,

*sudo apt-get install python-pip*

**Step 3,**

Install needed python libraries

Since there are a bunch of them, we developed a script to do this, find requirements.txt under the root folder of SmartAT source code then execute following commands in terminal,

*sudo pip install -r requirements.txt --use-mirrors*

In the coming three steps, you are going to install three databases needed by SmartAT, please be noticed that you can either install them on other PC or all on one as you wish, since you know SmartAT just needs theirs IP address.

**Step 4,**

Install mongodb,

Execute following commands in terminal,

*sudo apt-get install mongodb*

**Step 5,**

Install redis,

Execute following commands in terminal,

*sudo apt-get install redis-server*

**Step 6,**

Install memcached,

We strongly recommend you to install memcached on the PC which SmartAT will be running on for the sake of performance purpose.

Execute following commands in terminal,

*sudo apt-get install memcached*

**Start SmartAT**

Note: before any step in this chapter, please make sure that you have executed all the 6 steps in the latest chapter successfully

**Step 1,**

Start web server,

We provide two ways to start web server of SmartAT, find app.py under the root folder of SmartAT source code and

For testing and internal use purpose,

execute following commands in terminal,

*MONGODB\_URI=mongodb://localhost:27017 REDIS\_URI=redis://localhost:6379 MEMCACHED\_URI=localhost:11211 WEB\_PORT=8080 python app.py*

For end users,

execute following commands in terminal,

*MONGODB\_URI=mongodb://localhost:27017 REDIS\_URI=redis://localhost:6379/0 MEMCACHED\_URI=localhost:11211 gunicorn -k "geventwebsocket.gunicorn.workers.GeventWebSocketWorker" --workers=2 --bind=localhost:8080 app:app --pid /tmp/smart-web.pid*

**Step 2,**

Start worker server,

There are 3 ways to start worker server, execute following commands in terminal,

One node server,

*REDIS\_URI=redis://localhost:6379/0 celery worker --app=smartserver.worker:worker*

Concurrent pool server,

*REDIS\_URI=redis://localhost:6379/0 celery worker --app=smartserver.v2.worker:worker -P gevent -c 1000*

Multi nodes server(4 in this example if CPU of the host PC has 4 cores),

*REDIS\_URI=redis://localhost:6379/0 celery multi start 4 --app=smartserver.worker:worker -P gevent -l info -c:1-4 1000*

**Step 3,**

Start regular tasks,

execute following commands in terminal,

*REDIS\_URI=redis://localhost:6379/0 celery beat --app=smartserver.worker:worker --pid=/tmp/periodic\_task.pid --detach*

**Step 4,**

Start watcher daemon,

execute following commands in terminal,

*python sessionWatcher.py*

If you have installed the 3 databases in different PCs, please update "localhost" in the commands with corresponding IP address.

If you know these servers, you are welcomed to modify the starting commands but please be sure you know what you are doing.