

Python web.py 实现的简单监控系统

关键字: python web.py nginx memcached mysql uwsgi

www.scpman.com

由 scpman 出品

QQ:1028521938



默认登录用户: scpman

默认密码: scpman





开发语言是 python2.7.3(2.6 到 2.7 都可正常运行此项目，其他版本下未进行测试)+web.py
数据库用的是 mysql
web 前端用 nginx+uswgi
session 用 memcached
操作系统支持 win 和 linux
目前这套安装方法考虑到大家用 centos 的较多，是在 centos 6.4 i386（最小化系统）上安装的。

安装一些常用命令：yum -y install wget cc gcc make

第一部分：python 相关的安装开始

- 1、centos6.4 默认的 python 2.6.6 就用这个吧，无需升级
- 2、用 mysql 就得安装 py 的 mysql 插件：

```
yum search MySQL-python #查一下
```

```
yum install -y MySQL-python #安装一下
```

测试方法:

3、验证码那块用到了 python 的 PIL 图片处理库，所以得安装下

```
yum search python-imaging #查一下
```

```
yum -y install python-imaging #安装一下
```

4、安装 web.py 一个 python 的 web 框架

```
mkdir software #我们所有安装的软件包都放在这里
```

```
cd software/
```

```
wget http://webpy.org/static/web.py-0.37.tar.gz
```

```
tar -zxvf web.py-0.37.tar.gz
```

```
cd web.py-0.37
```

```
python setup.py install
```

5、安装一下 python 的时区模块

```
cd ..
```

```
wget https://pypi.python.org/packages/source/p/pytz/pytz-2008b.tar.gz --no-check-certificate
```

```
tar -zxvf pytz-2008b.tar.gz
```

```
cd pytz-2008b
```

```
python setup.py install
```

第一部分就安装完了，测试下我们是否安装成功？

```
# python #linux 命令行输入 python 就进入 python 命令行模式了
```

```
Python 2.6.6 (r266:84292, Feb 21 2013, 23:54:59)
```

```
[GCC 4.4.7 20120313 (Red Hat 4.4.7-3)] on linux2
```

```
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>> import web # import 是引入一个模块，web->web.py 没报错，成功
```

```
>>> import PIL #PIL ->图片处理，也没报错
```

```
>>> import MySQLdb #mysql 模块
```

```
>>> import pytz #时区模块,到此，python 相关的我们就安装完了
```

```
>>>exit() #退出 python 命令行模式，回到我们熟悉的 linux 命令行模式
```

第二部分：安装 mysql

```
#yum -y install mysql-server #一条命令安装完成
```

```
# /etc/init.d/mysqld start #启动服务
```

```
# mysql -uroot -p
```

```
Enter password: #默认是空密码
```

```
#我们设置下 mysql 的 root 密码
```

```
mysqladmin -u root password '你的新密码'
```

```
mysql 部分完成
```

第三部分：安装 memcached

```
#yum -y install memcached #安装 memcached
```

```
#yum -y install python-memcached.noarch #安装 python 插件
```

与此模块相关的代码例子，见我的博客：<http://www.scpman.com/article/show/31/?classid=4>

第 4 部分：安装 nginx+uwsgi

1、安装 nginx:

```
cd ..
```

```
wget http://nginx.org/download/nginx-1.5.4.tar.gz #来个最新的吧
```

```
tar -zxvf nginx-1.5.4.tar.gz
```

```
cd nginx-1.5.4
```

Nginx 编译需要 pcre 和 openssl 的支持，需要先安装下：

```
yum -y install pcre-devel openssl openssl-devel
```

```
./configure --prefix=/usr/local/nginx
```

```
make
```

```
make install
```

#/usr/local/nginx 的安装位置在这

2、安装 uwsgi

官网巨慢，时而打不开，可从我的博客下载：

```
cd ..
```

```
wget http://www.scpman.com/static/upload/uwsgi-1.9.13.tar.gz
```

```
tar -zxvf uwsgi-1.9.13.tar.gz
```

```
cd uwsgi-1.9.13
```

```
make #一下会生成个 uwsgi 命令
```

```
cp uwsgi /usr/local/bin/ #将命令拷贝到这（或者你自己喜欢的位置，给其执行权限就可以了）
```

```
#uwsgi
```

```
# whereis uwsgi
```

```
uwsgi: /usr/local/bin/uwsgi #查看下位置
```

3、配置文件的整合

这是我的 nginx.conf 配置文件内容，仅供参考

```
# vi /usr/local/nginx/conf/nginx.conf,将以下内容复制进去保存退出
```

```
#user nobody;
```

```
worker_processes 1;
```

```
error_log logs/error.log;
```

```
error_log logs/error.log notice;
```

```
error_log logs/error.log info;
```

```
pid logs/nginx.pid;
```

```
events {
```

```
    worker_connections 1024;
```

```
}
```

```
http {
```

```
    include mime.types;
```

```
    default_type application/octet-stream;
```

```

log_format main '$remote_addr - $remote_user [$time_local] "$request" '
                '$status $body_bytes_sent "$http_referer" '
                '"$http_user_agent" "$http_x_forwarded_for"';
access_log logs/access.log main;
sendfile on;
#tcp_nopush on;

#keepalive_timeout 0;
keepalive_timeout 65;
gzip on;
gzip_proxied any;
gzip_min_length 1024;
gzip_buffers 4 8k;
gzip_comp_level 3;
gzip_types text/plain text/css application/x-javascript application/javascript
application/xml;

```

include server_conf/*.conf; #mkdir -p /usr/local/nginx/conf/server_conf 新建立了个目录来放各个项目的对应 nginx 配置

```

}

```

```

#cd /usr/local/nginx/conf/server_conf

```

```

#vi web_jk.conf

```

```

server {
    listen 80;
    server_name localhost;
    error_page 503 /usr/local/nginx/html/503.html;
    access_log logs/access.log;
    location /static/ {
        root /usr/web_app/webjk_app/static/; #指向到我们项目的 static 目录:
        absolute_path
        rewrite ^/static/(.*)$ /$1 break;
    }
    location / {
        include /usr/local/nginx/conf/uwsgi_params;
        uwsgi_pass 127.0.0.1:8001;
    }
}

```

整合完成:

```

# /usr/local/nginx/sbin/nginx -t

```

```

nginx: the configuration file /usr/local/nginx/conf/nginx.conf syntax is ok

```

```

nginx: configuration file /usr/local/nginx/conf/nginx.conf test is successful

```

语法正常!

最后一步，将监控项目部署到当前环境中

1、代码：

注意 nginx 那个 static 那地方：

/usr/web_app/webjk_app/static/

所以我们的代码是放在/usr/web_app/webjk_app/下的

mkdir -p /usr/web_app/webjk_app/

cd /usr/web_app/webjk_app/

下载代码和 sql

wget http://www.scpman.com/static/upload/scpman_jk_1.0.tar

tar -zxvf scpman_jk_1.0.tar.gz

mv scpman_jk/* ./

rm -rf scpman_jk*

#到这代码放好了，导入库

mysql -uroot -p -e "create database jk_app;" #我们建立了个库叫 jk_app

Enter password: 输入你的密码

#将 sql 导入进库中

mysql -uroot -p jk_app < jk_app.sql --default-character-set=utf8

Enter password: 输入你的密码

#修改下代码项目的配置文件：

vi config.py

找到这一行

dbw = web.database(dbn='mysql', db='jk_app', user='root', pw='123456', host='127.0.0.1')

pw 换成你自己的数据库密码

接下来，启动所有服务：

#启动 memcached

memcached -d -m 256 -p 11211 -u root -l 127.0.0.1

#启动 mysqld

/etc/init.d/mysqld start

#启动 nginx

/usr/local/nginx/sbin/nginx

#启动后端的 uwsgi

cd /usr/web_app/webjk_app/;

find /usr/web_app/webjk_app/ -name "*.pyc" -type f |xargs rm;

ps auxww|grep uwsgi|grep -v "grep"|awk '{print \$2}'|xargs kill -9

sleep 1

ps auxww|grep uwsgi|grep -v "grep"

echo ok

uwsgi -s 127.0.0.1:8001 -w application -M -p 4 -t 30 -d uwsgi.log

我们自己弄一个启动的脚本：

#vi /usr/web_app/webjk_app/jk_app_start.sh

memcached -d -m 256 -p 11211 -u root -l 127.0.0.1

```
/etc/init.d/mysqld start
/usr/local/nginx/sbin/nginx
cd /usr/web_app/webjk_app/;
find /usr/web_app/webjk_app/ -name "*.pyc" -type f |xargs rm;
ps auxww|grep uwsgi|grep -v "grep"|awk '{print $2}'|xargs kill -9
sleep 1
ps auxww|grep uwsgi|grep -v "grep"
uwsgi -s 127.0.0.1:8001 -w application -M -p 4 -t 30 -d uwsgi.log
echo ok
#chmod +x /usr/web_app/webjk_app/jk_app_start.sh
/usr/web_app/webjk_app/jk_app_start.sh 下次直接运行这个脚本启动相关服务
```

此时在浏览器通过 ip 就可以访问到项目了

<http://你的ip>

就能看到开始时的那几张图了