### 主流部署方式

### Django+nginx+tornado

黄向昂

~~由于tornado并没有自带deamon运行，我们需要安装python-deamon来以服务方式启动tornado(用这个会有问题)~~

1. ~~从pip安装python-daemon:~~

|  |
| --- |
| ~~pip install python-daemon~~ |

1. 安装tornado：

|  |
| --- |
| pip install tornado |

1. 在项目目录 blog\_project底下建立一个 tornadoServer.py 文件，填入以下配置：

|  |
| --- |
| #/usr/bin/env python  from tornado.options import options, define, parse\_command\_line  from django.core.wsgi import get\_wsgi\_application  import tornado.httpserver  import tornado.ioloop  import tornado.web  import tornado.wsgi  #import daemon  import os, sys  SITE\_ROOT = os.path.dirname(os.getcwd())  PROJECT\_NAME = os.path.basename(os.getcwd())  sys.path.append( SITE\_ROOT )  os.environ['DJANGO\_SETTINGS\_MODULE'] = PROJECT\_NAME + '.settings'  define('port', type=int, default=8080)  class HelloHandler(tornado.web.RequestHandler):  def get(self):  self.write('Hello from tornado')  def main():  tornado.options.parse\_command\_line()  wsgi\_app = tornado.wsgi.WSGIContainer(  get\_wsgi\_application())  tornado\_app = tornado.web.Application(  [  ('/hello-tornado', HelloHandler),  ('.\*', tornado.web.FallbackHandler, dict(fallback=wsgi\_app)),  ])  server = tornado.httpserver.HTTPServer(tornado\_app)  server.listen(options.port)  tornado.ioloop.IOLoop.instance().start()  if \_\_name\_\_ == '\_\_main\_\_':  # with daemon.DaemonContext():  main() |

1. 为了方便以后启动、停止tornado，我们编写一个shell脚本来控制tornado，脚本放置于tornadoServer.py同样目录。

|  |
| --- |
| #!/bin/bash  serverName=$(ls|grep 'Server.py')  tornadoPorts='8000 8001 8002 8003'    function start {  for port in $tornadoPorts  do  nohup python $serverName --port=$port&  echo "Server: $serverName with port $port was Started!";  done  exit 1    }  function stop {  TORNADOS=$(ps ax |grep $serverName |grep -v grep| awk -F ' ' '{print $1}')  for pid in $TORNADOS  do  echo "Killing pid: $pid"  kill -9 $pid;  done  }  case "$1" in  start)  if ( start ); then  exit 0  else  exit 0;  fi  ;;  stop)  stop  ;;    restart)  stop  if ( start ); then  exit 0  else  exit 0;  fi  ;;  \*)  echo $"Usage: $0 {start|stop|restart}"  exit 1  esac |

1. 给予 tornado.sh 以执行权限：

|  |
| --- |
| Chomod +x tornado.sh |

这样，以后就可以到该脚本所在目录用 ./tornado.sh start 或者 ./tornado.sh stop 这样来启动和停止tornado的web服务。

1. 类似于gunicorn，我们安装Nginx：

yum -y install nginx

如果安装不了或者找不到nginx的源，则手动添加文件：

/etc/yum.repos.d/nginx.repo

在里面填入：

[nginx]

name=nginx repo

baseurl=http://nginx.org/packages/centos/6/$basearch/

gpgcheck=0

enabled=1

然后再执行一次yum -y install nginx

配置成开机启动，并手动启动nginx服务：

chkconfig nginx on

service nginx start

1. 修改nginx的配置/etc/nginx/nginx.conf：

|  |
| --- |
| user shawn;  worker\_processes 1;  error\_log /var/log/nginx/error.log;  pid /var/run/nginx.pid;  events {  worker\_connections 1024;  use epoll;  }  http {  # Enumerate all the Tornado servers here  upstream frontends {  server 127.0.0.1:8000;  server 127.0.0.1:8001;  server 127.0.0.1:8002;  server 127.0.0.1:8003;  }  include /etc/nginx/mime.types;  default\_type application/octet-stream;  access\_log /var/log/nginx/access.log;  keepalive\_timeout 65;  proxy\_read\_timeout 200;  sendfile on;  tcp\_nopush on;  tcp\_nodelay on;  gzip on;  gzip\_min\_length 1000;  gzip\_proxied any;  gzip\_types text/plain text/html text/css text/xml  application/x-javascript application/xml  application/atom+xml text/javascript;  # Only retry if there was a communication error, not a timeout  # on the Tornado server (to avoid propagating "queries of death"  # to all frontends)  proxy\_next\_upstream error;  server {  listen 80;  # Allow file uploads  client\_max\_body\_size 50M;  location ^~ /static/ {  root /home/shawn/blog\_project;  if ($query\_string) {  expires max;  }  }  location ^~ /media/ {  root /home/shawn/blog\_project;  if ($query\_string) {  expires max;  }  }  location = /favicon.ico {  rewrite (.\*) /static/favicon.ico;  }  location = /robots.txt {  rewrite (.\*) /static/robots.txt;  }  location / {  proxy\_pass\_header Server;  proxy\_set\_header Host $http\_host;  proxy\_redirect off;  proxy\_set\_header X-Real-IP $remote\_addr;  proxy\_set\_header X-Scheme $scheme;  proxy\_pass http://frontends;  }  }  } |

1. 修改settings.py 里的ALLOWED\_HOST:

|  |
| --- |
| ALLOWED\_HOSTS = ['localhost','example.com',’\*’] |

1. 禁用SELinux，编辑vi /etc/selinux/config 把里面的enforcing改成disabled 并重启。

参考：

<https://gist.github.com/mignev/6537140>  
<http://www.tornadoweb.org/en/stable/wsgi.html?highlight=django>

<http://www.tornadoweb.org/en/stable/guide/running.html>