Robert's blog

Tuesday, March 27, 2012

Gunicorn + Nginx - a much better way to deploy your Django website

I've been a lighttpd + FastCGI django user for a long time.
Now there's a better solution -> gunicorn.

What is a gunicorn?

> Gunicorn 'Green Unicorn' is a Python WSGI HTTP Server for UNIX. It's a pre-fork worker model ported from Ruby's Unicorn project. The Gunicorn server is broadly compatible with various web frameworks, simply implemented, light on server resources, and fairly speedy.

If you are a fastcgi user, please check gunicorn asap. I'm sure you will get much more from your hardware :-)
Ps: Note that you get the most if database is not the bottleneck. That's why you should use cache everywhere if possible.

Here's my script for gunicorn + daemontools for supervising:

```
#!/bin/sh

# SITE_PATH is a directory where settings.py is stored
SITE_PATH=/var/www/django/projects/my_project/; export SITE_PATH
USER=www-data
TCP=127.0.0.1:8888

exec setuidgid $USER /usr/bin/gunicorn_django -b ${TCP} -w 3 --max-requests=1000 ${SITE_PATH}settings.py 2>&1
```

Seems like wsgi is also recommended with the new 1.4 version of Django

> If you're new to deploying Django and/or Python, we'd recommend you try mod_wsgi first. In most cases it'll be the easiest, fastest, and most stable deployment choice.

Check django site for more information.

UPDATE:
Forgot to mention that I've switched from lighttpd to Nginx.
You can use the script above with the following nginx config:

```
server {
    listen   xx.xx.xx.xx:80;
    server_name domain.com www.domain.com;
    root /var/www/django/projects/my_project/public;
    access_log /var/log/nginx/domain-access.log;

    rewrite ^/admin(.*) https://domain.com/admin$1 permanent;

    location = /favicon.ico { access_log off; log_not_found off; }
    location /media {
        root /var/www/django/projects/my_project/django/contrib/admin;
    }
    location /static {
        root /var/www/django/projects/my_project/site_media;
        access_log off;
    }
    location /site_media {
        root /var/www/django/projects/my_project;
    }
    location / {
        proxy_pass_header Server;
        proxy_set_header Host $http_host;
        proxy_redirect off;
        proxy_set_header X-Forwarded-For  $remote_addr;
        proxy_set_header X-Scheme $scheme;
        proxy_connect_timeout 10;
```

Pages

- Home
- About me

Blog Archive

- ► 2015 (1)
- ► 2014 (2)
- ► 2013 (2)
- ▼ 2012 (10)
  - ► 09 (3)
  - ► 06 (2)
  - ► 04 (1)
  - ▼ 03 (2)
    - Gunicorn + Nginx - a much better way to deploy you...
    - Django HTML5 date input widget with i18n support a...
  - ► 02 (2)
- ► 2009 (3)
- ► 2008 (6)
- ► 2007 (4)

Ciekawe linki

- Ruch Wolność i Godność
- TaiChi, KungFu, Qigong - YMAA Gdynia

```
        proxy_read_timeout 10;
        proxy_pass http://127.0.0.1:8888/;
    }

}
```

There's a trick you need to apply to get the real ip of the request source as seen in the line:

```
proxy_set_header X-Forwarded-For  $remote_addr;
```

Here's a code I'm using:

```
class XForwardedForMiddleware(object):
       def process_request(self, request):
        if "HTTP_X_FORWARDED_FOR" in request.META:
            ip = request.META["HTTP_X_FORWARDED_FOR"]
            if ip.startswith('::ffff:'):
                ip = ip[len('::ffff:'):]
            request.META["REMOTE_ADDR"] = ip
            request.META["REMOTE_HOST"] = None
```

Add **XForwardedForMiddleware** to your settings.py **MIDDLEWARE_CLASSES** setting.

Autor: Robert o 12:30 PM

Etykiety: django, gunicorn, nginx, performance, python, wsgi

5 comments:

Anonymous said...

According to benchmarks, Gunicorn does not provide sufficient performance benefits over mod_wsgi. If you are already using nginx, I'd use uWSGI which is faster than Gunicorn and better integrated into nginx.

March 28, 2012 at 1:04 AM

Anonymous said...

Thanks for info. I know about uWSGI but have not tested it yet on my application. Will do some tests and publish the result.

March 28, 2012 at 1:25 AM

huxley said...

Yeah, according to benchmarks that are the equivalent of hello world.

In real life, with a real web app, the wsgi server is not a significant performance drag.

Even the uWSGI guys say that performance is not a factor (they prefer that people pick uWSGI because they need flexibility and extendibility rather than performance).

The real benefit to using the nginx proxy to gunicorn is that it needs far less memory than even a tuned mod_wsgi/Apache install.

March 28, 2012 at 3:07 PM

Anonymous said...

Gevent or fapws are faster !

http://nichol.as/benchmark-of-python-web-servers
http://www.fapws.org/

Note: FAPWS support django out of the box !

April 7, 2012 at 9:08 AM

Anonymous said...

http://nichol.as/benchmark-of-python-web-servers
is over two year old.

June 14, 2012 at 3:01 AM

Post a Comment

Newer Post Home Older Post