🗎 Nginx, PHP and a PHP FastCGI daemon init script

- June 11, 2007

One of the things us ex-Apache httpd or ex-Lighttpd users have to get used to when installing nginx is how there isn't built-in FastCGI support for process spawning. While these means nginx is more lightweight and faster, it does mean that you have to manage your FastCGI processes yourself.

Having just upgraded to Ubuntu Feisty Fawn, I set to installing nginx and tried to get WordPress (which needs PHP, of course) running on it. Thanks to the new nginx package in the Feisty universe repository, getting nginx up and PHP (CGI version) and running is really straightforward. There're lots of tutorials to help you out as well.

So that's that. The part that bugged me though, was how the tutorials always provided a script to start your PHP FastCGI processes, but never provided a way to ensure your FastCGI processes started up on a reboot (i.e. an init script). Once again, Google doesn't disappoint, turning up this php-cgi init script. Tweak the init script a little, put it in /etc/init.d/php-fastcgi, `sudo chmod +x /etc/init.d/php-fastcgi`, run `sudo update-rc.d php-fastcgi defaults` and place the configuration for the script in /etc/default/php-fastcgi, and you're done. You now have a PHP FastCGI init script that spawns and kills your PHP FastCGI processes.

Note: Lighttpd does come with a `spawn-fcgi` binary that does the same, but I'm having trouble finding a suitable init script for spawn-fcgi (only the Gentoo emerge seems to have the init script). Still, this script works and I didn't really want to install lightty on my starving VPS.

The init script: /etc/init.d/php-fastcgi

```
#! /bin/sh
### BEGIN INIT INFO
# Provides:          php-fastcgi
# Required-Start:    $all
# Required-Stop:     $all
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: Start and stop php-cgi in external FASTCGI mode
# Description:       Start and stop php-cgi in external FASTCGI mode
### END INIT INFO

# Author: Kurt Zankl <[EMAIL PROTECTED]>

# Do NOT "set -e"


PATH=/sbin:/usr/sbin:/bin:/usr/bin
DESC="php-cgi in external FASTCGI mode"
NAME=php-fastcgi
DAEMON=/usr/bin/php-cgi
PIDFILE=/var/run/$NAME.pid
SCRIPTNAME=/etc/init.d/$NAME
PHP_CONFIG_FILE=/etc/php5/cgi/php.ini

# Exit if the package is not installed
[ -x "$DAEMON" ] || exit 0

# Read configuration variable file if it is present
[ -r /etc/default/$NAME ] && . /etc/default/$NAME

# Load the VERBOSE setting and other rcS variables
. /lib/init/vars.sh

# Define LSB log_* functions.
# Depend on lsb-base (>= 3.0-6) to ensure that this file is present.
. /lib/lsb/init-functions

# If the daemon is not enabled, give the user a warning and then exit,
# unless we are stopping the daemon
if [ "$START" != "yes" -a "$1" != "stop" ]; then
        log_warning_msg "To enable $NAME, edit /etc/default/$NAME and set START=yes"
        exit 0
fi

# Process configuration
export PHP_FCGI_CHILDREN PHP_FCGI_MAX_REQUESTS
```

Config file for init script (the init script looks for this): /etc/default/php-fastcgi

```
START=yes

# Which user runs PHP? (default: www-data)

EXEC_AS_USER=www-data

# Host and TCP port for FASTCGI-Listener (default: localhost:9000)

FCGI_HOST=localhost
FCGI_PORT=9000

# Environment variables, which are processed by PHP

PHP_FCGI_CHILDREN=4
PHP_FCGI_MAX_REQUESTS=1000
```

nginx config file: /etc/nginx/nginx.conf

```
location ~ \.php$ {
  fastcgi_pass   127.0.0.1:9000;
  fastcgi_index  index.php;
  fastcgi_param  SCRIPT_FILENAME  /var/www/blog.codefront.net$fastcgi_script_name;
  include        /etc/nginx/fastcgi.conf;
}
```

My fastcgi.conf file (I'm not sure I need everything here...)

```
fastcgi_param  QUERY_STRING        $query_string;
fastcgi_param  REQUEST_METHOD      $request_method;
fastcgi_param  CONTENT_TYPE        $content_type;
fastcgi_param  CONTENT_LENGTH      $content_length;

fastcgi_param  SCRIPT_NAME         $fastcgi_script_name;
fastcgi_param  REQUEST_URI         $request_uri;
fastcgi_param  DOCUMENT_URI        $document_uri;
fastcgi_param  DOCUMENT_ROOT       $document_root;
fastcgi_param  SERVER_PROTOCOL     $server_protocol;

fastcgi_param  GATEWAY_INTERFACE   CGI/1.1;
fastcgi_param  SERVER_SOFTWARE     nginx;

fastcgi_param  REMOTE_ADDR         $remote_addr;
fastcgi_param  REMOTE_PORT         $remote_port;
fastcgi_param  SERVER_ADDR         $server_addr;
fastcgi_param  SERVER_PORT         $server_port;
fastcgi_param  SERVER_NAME         $server_name;

# PHP only, required if PHP was built with --enable-force-cgi-redirect
fastcgi_param  REDIRECT_STATUS     200;
```

Save to del.icio.us (15 saves, tagged: nginx php fastcgi) • Digg This!

Posted by Chu Yeow at June 11, 2007 SGT | 📄 Permalink | 💬 | Category: Open Source, Operating Systems, Ruby on Rails

9 Comments & TrackBacks (Add yours)

The paper doll icon that precedes each comment is an idea conceived by Vanessa Tan.

[...] Nginx, PHP and a PHP FastCGI daemon init script (tags: php web nginx fastcgi sysadmin) [...]

Posted by: *links for 2007-06-13 « Bloggitation* on June 13, 2007 8am

I haven't gotten it working yet, but I can make a couple corrections:

It's not "/etc/defaults/php-fascgi", it's "/etc/default/php-fascgi".

Beware when copying and pasting the above scripts. Double quotes and double hyphens get all messed up.

I'm currently stuck at "/etc/init.d/php-fastcgi start". It's just freezing without an error messages.

Posted by: *Shannon -jj Behrens* on June 14, 2007 3pm

Thanks for the corrections. I've fixed the typo, though I can't figure out how to stop WordPress from "correcting" the quotes and hyphens in the code.

So it's just freezing and you have to kill the script? It's possible with the double hyphens and quotes madness there was a typo somewhere.

Posted by: *Chu Yeow* on June 14, 2007 3pm

[...] So, it's been a little bit quiet on this blog for the past few days. The main reason is that it's been on the move from the hosting provided by our ISP (PlusNet) to my SliceHost slice. Things would have been much quicker had I not decided to switch Apache for the increasingly popular (due to its low memory usage) nginx. As is the beauty of a virtual server, with the click of a mouse I had a clean Dapper Drake install to start with. Rather than install nginx from source, I decided to upgrade to Feisty Fawn and install via apt. This means losing the long term support of Dapper but, hey, its my server and I'll just have to upgrade again when necessary. From there, getting an nginx, mysql, Rails and PHP setup is well documented, on top of which I added an init script for FastCGI. [...]

Posted by: *David Currie » Blog Archive » Changing servers* on July 14, 2007 1am

Does a Fedora/RH equivalent of this script exist? I've been tweaking this to get it working on my Fedora machine, but have not had luck yet.

Thanks,
Brian

Posted by: *Brian Acheson* on July 22, 2007 6am

yes, fedora does have it, you could try: lighttpd-fastcgi it should give desired effects.

Posted by: *lejeczek* on August 16, 2007 8pm

i actually use ubuntu's "upstart" init replacement, and spawn-fcgi from lighttpd

```
start on runlevel 2
start on runlevel 3
start on runlevel 4
start on runlevel 5
```

stop on shutdown

env PHP_FCGI_MAX_REQUESTS=125

respawn
exec /usr/bin/spawn-fcgi -u user -g group -C 5 -n -a 127.0.0.1 -p 1234 -f /usr/local/bin/php-cgi

then you can use "start $job" "stop $job" when you need it, and add that to your init script for starting/stopping the webserver (start engines prior to webserver starting, end engines after webserver ends)

works for me so far...

Posted by: *mike503* on August 21, 2007 8am

hi,

difficult to make your scrip work. after correcting the quotes i have that new one:

/etc/init.d/php-fastcgi: 55: Syntax error: "||" unexpected. could you post a link to the raw script as .txt for instance? would be great. thank you

Posted by: *osiris* on August 28, 2007 12am

Go to this page to download the script files above. Just follow the instructions step by step.

Posted by: *Gerry Kirk* on August 31, 2007 9pm