**Crontab定时任务之执行PHP脚本**

假定定时执行的文件在/www/task/文件夹中，则

1. 创建定时任务需要执行的文件task.php

#/usr/bin/env php //task.php文件头信息 - 系统兼容性

**<?php**

//开始编辑PHP执行脚本

**?>**

2. 修改文件所属文件权限：

chown -R www:www /www/task/ //将任务执行的脚本定义为www用户有权限执行

chmod -R 744 /www/task/

3. 打开crontable定时任务编辑器

crontab -u www -e //编辑www用户的定时任务文件

4. 编辑定时任务文件

格式： 运行脉搏-即时间间隔 要用什么程序执行定时任务的脚本 脚本所在地址路径 每次执行后的结果写入到指定日志文件中

注： 路径需要写绝对路径，且日志写入文件需要有”写"的权限

1. 如果不需要保存任务执行日志，则

\*/1 \* \* \* \* /usr/local/php/bin/php /www/task/task.php > /dev/null

2. 如果需要保存任务执行日志，则

\*/1 \* \* \* \* /usr/local/php/bin/php /www/task/task.php >> /var/log/task/task.log

5. 自动执行PHP文件格式如下# crontab -e  
 第一种： 执行时间 php路径 PHP文件路径  
 比如： 0 \*/2 \* \* \* /usr/local/php/bin/php /var/www/object/Application/test.php  
  
 第二种：直接执行浏览器访问的路径  
 如果你的PHP脚本可以通过URL触发，你可以使用lynx或curl或wget来配置你的Crontab。  
 一.

下面的例子是使用Lynx文本浏览器访问URL来每小时执行PHP脚本。Lynx文本浏览器默认使用对话方式打开URL。但是，像下面的，我们在lynx 命令行中使用-dump选项来把URL的输出转换来标准输出。  
0 \*/2 \* \* \* lynx -dump http://ip地址或者域名/Api/robot （使用该命令前 确定安装lynx）  
// 比如：00 \* \* \* \* lynx -dump http://www.jb51.net/myscript.php  
 二.

下面的例子是使用CURL访问URL来每5分执行PHP脚本。Curl默认在标准输出显示输出。使用”curl -o”选项，你也可以把脚本的输出转储到临 时文件。  
0 \*/2 \* \* \* /usr/bin/curl http://ip地址或者域名/Api/robot  
//比如： \*/5 \* \* \* \* /usr/bin/curl -o temp.txt http://www.jb51.net/myscript.php  
 三.

下面的例子是使用WGET访问URL来每10分执行PHP脚本。-q选项表示安静模式。”-O temp.txt”表示输出会发送到临时文件。  
\*/10 \* \* \* \* /usr/bin/wget -q -O temp.txt http://www.jb51.net/myscript.php

5. 编辑完成后，重启crond服务

service crond restart //CentOS6.x

systemctl restart crond //CentOS7.x

6. 任务列表查看：

crontab -u www -l //查看www用户下的任务列表

7. 删除任务

1. 删除指定任务： crontab -u www -e //编辑，删除指定任务即可

2. 删除所有任务： crontab -u www -r //慎用，会删除www用户所有任务

删除任务后，记得重启crond

**crontab无法执行php的解决方法**

**一、php文件有没有执行权限**

代码如下:

[root@linux cron]# ls -al |grep del

 -rwxr-xr-x  1 zwh  ftpgroup  494 10-20 16:42 del\_redis.php   
如果没有X，说明没有执行权限，当然也执行不了。如果你不知道你登录的用户所在的组是不是有权限，直接给所有组都加上权限就行了。方法如下：

代码如下:

[root@linux cron]# chmod +x ./del\_redis.php

**二、如果有执行权限了，还是执行不了**

解决方法如下：

1. 在php程序里面写log(例如：error\_log()这个函数)，这也是必须的，因为crontab定时执行，要是没有log，你怎么知道程序执行的结果怎么样。这样我们可以通过log来查看crontab执行的情况。

2. 也可以直接在crontab里面，将执行的结果输出一个文件当中。然后在这个文件当中查看执行情况。例如：

复制代码代码如下:

\*/10 \* \* \* \* /usr/local/php/bin/php /var/www/cron/del\_redis.php >> /home/zhangy/cron.txt

3. 用crontab来直接执行php【直接访问php文件】，是不走apache,nginx，没有什么http协议，所有get,post传参数根本不起作用，并且还会报错，所以$\_SERVER,$\_ENV这类变量根本用不了。所以检查一下php代码中有没有这类变量，如果有拿掉。

4. php的相对路径问题

因为有思维定势在做怪，这个问题也是最容易忽视的。

代码如下:

include\_once'./mysql.php';  
当php代码中，用的是相对路径时，只有进入到那个目录下执行/usr/local/php/bin/php /var/www/cron/level\_rank.php才能生效。这个问题我遇到过至少二次，但是再遇到时还是想不起来。

解决方法如下:

代码如下:

\*/10 \* \* \* \* cd /var/www/cron && /usr/local/php/bin/php /var/www/cron/level\_rank.php  
或者 在php代码中用绝对路径