



链滴

# Nginx Smooth Upgrade

作者: [xfell](#)

原文链接: <https://ld246.com/article/1549033530311>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<blockquote>

<p>Nginx 在不停掉老进程的情况下 升级版本启动新进程 老进程负责处理仍然没有处理完的请求，进程接替老进程开始接收请求</p>

</blockquote>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">./nginx -V # 查看当前版本
</span></span><span class="highlight-line"><span class="highlight-cl">cp /usr/local/serve
/nginx/sbin/nginx nginx.old # 备份旧版本的可执行文件
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"># 下载最新Nginx
编译，要注意的是只执行make时 不能紧接着 make install 这样会覆盖你的二进制文件 这里只升级
本
</span></span><span class="highlight-line"><span class="highlight-cl">wget http://nginx.
rg/download/nginx-1.14.0.tar.gz
</span></span><span class="highlight-line"><span class="highlight-cl">tar xf nginx-1.14.0.
ar.gz
</span></span><span class="highlight-line"><span class="highlight-cl">cd nginx-1.14.0/
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">./configure --prefi
=/usr/local/server/nginx --user=nginx --group=nginx --with-http_ssl_module --with-http_stu
_status_module --with-http_realip_module # 最后一个模块是新增的 获取后端真实IP地址
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">make # 千万不要
ake install
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"># 拷贝新的可执行
件到旧版本的目录下
</span></span><span class="highlight-line"><span class="highlight-cl">cp -rfp nginx-1.14
0/objs/nginx /usr/local/server/nginx/sbin/
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"># 检查配置文件有
有错误
</span></span><span class="highlight-line"><span class="highlight-cl">./nginx -t
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"># 发送信号给旧mas
er进程 接着启动新进程
</span></span><span class="highlight-line"><span class="highlight-cl">kill -USR2 `cat /usr
local/nginx/logs/nginx.pid`
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"># 发送WINCH给
master 让他优雅的退出 (因为发送USR2信号之后 旧master进程会重名为 nginx.pid.oldbin)
</span></span><span class="highlight-line"><span class="highlight-cl">kill -WINCH `cat /
sr/local/nginx/logs/nginx.pid.oldbin`
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"># 当旧master进程
受完所有之前的剩余请求之后 就可以发送QUIT让其退出即可
</span></span><span class="highlight-line"><span class="highlight-cl">kill -QUIT `cat /usr
local/nginx/logs/nginx.pid.oldbin`
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">#如果升级失败 新
aster进程没有接受请求 可以用下面2种方式快速回退
</span></span><span class="highlight-line"><span class="highlight-cl"># 发送HUP信号给
master进程。旧master进程会在不重新读取配置文件的情况下直接启动新的worker进程。然后就可
发送QUIT信号给新master进程，使所有新进程可以优雅退出。
</span></span>></pre>
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"> kill -HUP 15023
</span></span><span class="highlight-line"><span class="highlight-cl"> kill -QUIT 15130
</span></span><span class="highlight-line"><span class="highlight-cl">#发送TERM信号
新master进程。新master进程发送信号关闭所有的worker进程（如果因为某些原因新的进程没有退
，就得发送KILL信号强制退出了）。当新master进程退出后，旧master进程会自动启动它的worker
程。
</span></span><span class="highlight-line"><span class="highlight-cl"> kill -TERM 1513

</span></span><span class="highlight-line"><span class="highlight-cl"> kill -9 15130
</span></span></code></pre>
```