

nginx增加模块 平滑升级

①备份nginx

```
rsync -azv nginx/ nginx.bak/ --exclude=logs/*
```

②重新进行nginx安装步骤中configure之前的步骤

见笔记"nginx安装步骤"

③获取nginx当前的模块配置

```
./nginx -V
```

④重新进行configure

将③中获得到的nginx当前模块配置加上（不能少），另外，再加上本地升级需要添加的模块即可，如

```
./configure --prefix=/usr/local/nginx \  
--with-http_stub_status_module \  
--with-http_ssl_module \  
--with-http_gzip_static_module \  
--add-module=/usr/local/src/nginx/nginx_upstream_jvm_route \  
--add-module=/usr/local/src/nginx/nginx_upstream_check_module-master/ \  
--add-module=/usr/local/src/nginx/nginx-sticky-module-1.1 \  
--add-module=/usr/local/src/nginx/echo-nginx-module-0.51 \  
--add-module=/usr/local/src/nginx/nginx_cache_purge-2.1 \  
--add-module=/usr/local/src/nginx/nginx_slowfs_cache-1.10
```

⑤make，切忌不能make install（如果make install了就把nginx配置文件都覆盖了）

```
make
```

⑥替换新生成的nginx文件

```
#---先备份sbin/nginx执行文件  
cp -a /mall/web/nginx/sbin/nginx{,.bak}  
#---覆盖，需要在安装文件目录下执行  
\cp objs/nginx /mall/web/nginx/sbin/ -f
```

⑦平滑升级

```
#---./logs/nginx.pid为旧的nginx进程的pid文件  
kill -USR2 $(cat ../logs/nginx.pid)  
#---执行完之后旧的pid文件会自动重命名为nginx.pid.oldbin  
#---并且能够看到同时出现两个nginx主进程（一新一旧）以及各自的子进程  
#---杀死旧的nginx进程  
kill -QUIT $(cat ../logs/nginx.pid.oldbin)  
#---查看模块是否添加成功  
./nginx -V
```

⑧确认模块添加成功

```
#---这种方式也可以，但是是会造成一定时间的停机的  
#---这里必须使用stop，不可以直接reload，直接reload不会生效  
./nginx -s stop  
./nginx  
./nginx -V
```