一下内容转载于<https://www.jianshu.com/p/39b476e808d8>

**前言**

手头上的服务器程序1.0版本终于要进入线上测试阶段了😮。兴致冲冲的分了一台公司的系统为CentOS7.+的服务器。1、部署上环境。2、打开终端开始跑程序 "nohup dotnet XXX.dll &"。3、关闭终端。。。4、卧槽😮,我的后台运行的程序呢!?  
本人Linux小白,又尝试了几种后台运行程序的方案，全部以失败告终😭😭😭。  
然后在搜索解决方案的过程中发现了很多守护进程工具的东西,最终找到了一个最简单的工具并学习使用了它, [Supervisor](http://supervisord.org/) 。

[**进程管理工具(Supervisor)**](http://supervisord.org/introduction.html) **简介**

[Supervisor](http://supervisord.org/introduction.html#overview)是用Python开发的一个client/server服务，是Linux/Unix系统下的一个进程管理工具，不支持Windows系统。它可以很方便的监听、启动、停止、重启一个或多个进程。用Supervisor管理的进程，当一个进程意外被杀死，supervisort监听到进程死后，会自动将它重新拉起，很方便的做到进程自动恢复的功能，不再需要自己写shell脚本来控制。

不使用守护进程会出现的三个问题：

* 1、ASP.NET Core应用程序运行在shell之中，如果关闭shell则会发现 ASP.NET Core程序被关闭，从而导致应用无法访问，这种情况当然是我们不想遇到的，而且生产环境对这种情况是零容忍的。
* 2、如果 ASP.NET Core进程意外终止那么需要人为连进shell进行再次启动，往往这种操作都不够及时。
* 3、如果服务器宕机或需要重启，我们则还是需要连入shell进行启动。

为了解决这些问题，我们需要有一个程序来监听 ASP.NET Core 应用程序的状况。并在应用程序停止运行的时候立即重新启动。

**Supervisor安装与配置**

**1、安装Python包管理工具([easy\_install](https://pypi.org/project/setuptools/" \t "_blank))**

yum install python-setuptools

**2、安装Supervisor**

easy\_install supervisor

**3、配置Supervisor应用守护**

a) 通过运行echo\_supervisord\_conf程序生成supervisor的初始化配置文件，如下所示：

mkdir /etc/supervisor

echo\_supervisord\_conf > /etc/supervisor/supervisord.conf

然后查看路径下的supervisord.conf。在文件尾部添加如下配置。

...

;[include]

;files = relative/directory/\*.ini

;conf.d 为配置表目录的文件夹，需要手动创建

[include]

files = conf.d/\*.conf

b) 为你的程序创建一个.conf文件，放在目录"/etc/supervisor/conf.d/"下。

[program:MGToastServer] ;程序名称，终端控制时需要的标识

command=dotnet MGToastServer.dll ; 运行程序的命令

directory=/root/文档/toastServer/ ; 命令执行的目录

autorestart=true ; 程序意外退出是否自动重启

stderr\_logfile=/var/log/MGToastServer.err.log ; 错误日志文件

stdout\_logfile=/var/log/MGToastServer.out.log ; 输出日志文件

environment=ASPNETCORE\_ENVIRONMENT=Production ; 进程环境变量

user=root ; 进程执行的用户身份

stopsignal=INT

c) 运行supervisord，查看是否生效

supervisord -c /etc/supervisor/supervisord.conf

ps -ef | grep MGToastServer

成功后的效果：

ps 如果服务已启动，修改配置文件可用“supervisorctl reload”命令来使其生效

**4、配置Supervisor开机启动**

a) 新建一个“supervisord.service”文件

# dservice for systemd (CentOS 7.0+)

# by ET-CS (https://github.com/ET-CS)

[Unit]

Description=Supervisor daemon

[Service]

Type=forking

ExecStart=/usr/bin/supervisord -c /etc/supervisor/supervisord.conf

ExecStop=/usr/bin/supervisorctl shutdown

ExecReload=/usr/bin/supervisorctl reload

KillMode=process

Restart=on-failure

RestartSec=42s

[Install]

WantedBy=multi-user.target

b) 将文件拷贝至"/usr/lib/systemd/system/supervisord.service"

c) 执行命令

systemctl enable supervisord

d) 执行命令来验证是否为开机启动

systemctl is-enabled supervisord

**配置完成啦.**

**常用的相关管理命令**

supervisorctl restart <application name> ;重启指定应用

supervisorctl stop <application name> ;停止指定应用

supervisorctl start <application name> ;启动指定应用

supervisorctl restart all ;重启所有应用

supervisorctl stop all ;停止所有应用

supervisorctl start all ;启动所有应用

**Talk**

[将ASP.NET Core应用程序部署至生产环境中（CentOS7）](https://www.cnblogs.com/ants/p/5732337.html)

[Supervisor安装与配置（Linux/Unix进程管理工具）](https://blog.csdn.net/xyang81/article/details/51555473)

作者：Replay923  
链接：https://www.jianshu.com/p/39b476e808d8  
来源：简书  
著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。