



链滴

docker 容器的退出策略 (转载)

作者: [WongKai42817](#)

原文链接: <https://ld246.com/article/1642748407658>

来源网站: [链滴](#)

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



停止运行中的 docker 容器并正确的终止其中的程序。

可能会遇到的场景-

场景一：容器的程序提供 Http 方式的服务，负责处理各种 Http requests 并返回结果。假如希望在容器被停掉的时候，能够让程序有时间把已经在处理中的请求继续处理完毕，并返回结果给客户端

场景二：打包在容器中的程序，负责写入到某个数据文件，希望程序能够在容器被停掉的时候，时间把内存中缓存的数据持久化到存储设备中，防止数据丢失

场景三：比如微服务架构中，服务发现机制，每个微服务在启动后，会主动把自己的地址信息注到服务发现模块。而在容器被停掉时候，微服务需要同步从服务发现模块中注销自己，防止从 API Gateway 过来的请求被错误的路由到已经被停掉的微服务

docker stop 与 docker kill 的区别

docker stop



当使用 docker stop 命令来停掉容器的时候，docker 会默认容器中应用程序有十秒的时间以终止运行。

在 docker stop 命令执行后，会先向容器中 PID 为 1 的进程发送系统信号 SIGTERM，然后等容器中的应用程序终止进程，等待时间到达设定超时时间，或者默认 10s，会继续发送 SIGKILL 的信号强行 kill 掉进程。

在容器中的应用程序，可以选择忽略和不处理 SIGTERM 信号，但是达到超时时间，会被系统强制 kill 调，SIGKILL 信号是发往系统内核，应用程序没有机会处理。但是使用 docker stop 命令时候，以控制超时时间，比如设置为 20 秒超时：

```
docker stop --time=20 container_name
```

docker kill



默认情况下，docker kill 会直接发出 SIGKILL 的系统信号，一强行终止容器中程序的运行。通查看 docker kill 帮助文档，可以看到，除了默认发送 SIGKILL 信号，还允许发送一些自定义的系统号。

比如：希望向 docker 中程序发送 SIGINT 信号，我们可以这样实现：

```
docker kill --signal=SIGINT container_name
```

和 docker stop 不同的地方在于，docker kill 没有任何超时设置，会直接发送 SIGKILL 信号，及用户通过 signal 参数指定的其他信号。

docker stop 命令类似于 kill 命令，两者都是发送系统信号 SIGTERM。docker kill 更像 linux 的 kill -9，或者是 kill -SIGKILL 命令，用来强行终止进程。

总结

docker kill 适合用来清醒终止程序并实现快速停止容器。

生产环境停止 (gracefully shutdown)，docker stop 才比较合适。

我们可以让程序在接收到 SIGTERM 信号后，有一定的时间处理、保存程序执行现场，优雅退出。

文章来源：

<https://ld246.com/forward?goto=https%3A%2F%2Fwww.jb51.net%2Farticle%2F96617.htm> target="_blank" rel="nofollow ugc">docker 容器如何优雅的终止详解_docker_脚本家 (jb51.net)

<https://ld246.com/forward?goto=https%3A%2F%2Fwww.jb51.net%2Farticle%2F111945.htm> target="_blank" rel="nofollow ugc">linux kill 命令详解_Linux_脚本之家 (jb51.net)