

# 对讲服务器部署指南(docker 方式)

北京滔达科技有限公司 www.allptt.com

## 一、 硬件环境

服务器硬件至少双核 CPU，2GB 以上 RAM

以 1000 用户并发为例，建议最低配置：

服务器硬件至少双核 CPU，2GB 以上 RAM

100GB 以上硬盘

5Mbps 以上出口带宽

更多在线用户，基本成比例增加配置。其中出口带宽根据实际对讲并发数量适当调整。

## 二、 软件环境

Docker 运行要求要求 64 位 Linux 或 Windows 操作系统，且支持 docker 部署。

如无特殊要求，Linux 建议 Ubuntu18.04 以上或 centOS 7 以上。

Windows 系统要求详见 [Install on Windows | Docker Documentation](#) 的 System requirements 部分。建议选择 WSL 2 后端方式，操作系统及配置应满足以下要求：

●64 位处理器，4GB 以上内存。

●BIOS 设置里支持虚拟化。(如何查看是否已支持虚拟化? Windows 10/11 打开任务管理器(快捷键 Ctrl+Shift+ESC) 点击性能——CPU, 查看左下角虚拟化是否开启)

●支持的操作系统：

1) Windows 10 64 位：家庭或专业版的 21H1 (build 19043) 及以上版本，或企业版/教育版的 20H2 (build 19042) 及以上版本。

2) Windows 11 64 位：家庭或专业版的 21H2 及以上版本，或企业版/教育版的 21H2 及以上版本。

3) Windows server 2016 及以上版本。

●操作系统-启用或关闭系统功能里，开启“适用于 Windows 的 Linux 子系统”和“虚拟机平台”：



●打开 Microsoft Store, 搜索 Ubuntu 并下载, 建议 Ubuntu 18.04 或 Ubuntu 20.04。

## 三、 Docker 软件安装

Ubuntu 安装 docker:

```
sudo apt install docker.io
```

CentOS 安装 docker:

```
yum install -y docker-io
```

安装完成后, 执行 `docker -v` 有版本信息输出说明成功。

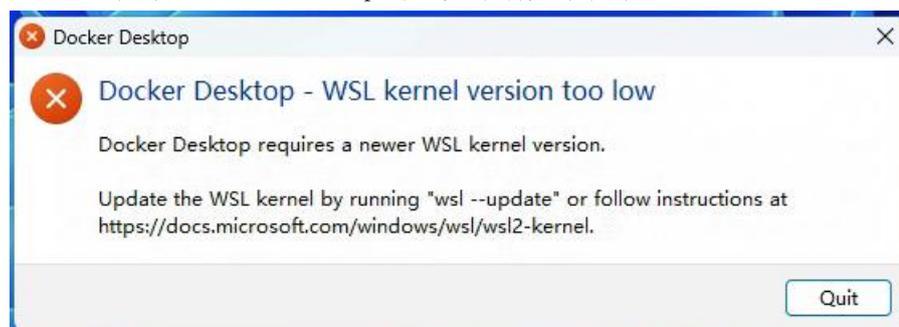
Windows 安装 docker:

浏览器访问 <https://www.docker.com/products/docker-desktop/>, 下载并安装。

打开命令行, 运行 `docker -v`, 有版本信息输出表明安装成功:

```
D:\>docker -v
Docker version 20.10.21, build baedalf
```

Windows 启动 Docker desktop 时, 如果有如下提示:



则打开命令行, 运行 `wsl --update`

#### 四、 端口要求

对讲服务器默认端口 (必选): 59638 (TCP 和 UDP)、59639 (TCP)

Web 后台管理默认端口 (一般必选): 8080, 8081 (TCP)

文件服务器默认端口 (一般必选): 59680 (TCP), 8443 (TCP)

直播服务器默认端口 (可选): 1935 (TCP), 1985 (TCP), 1990 (TCP), 8000 (UDP), 10080 (UDP)

网页对讲端口 (可选): 443 (TCP)

音视频通话 (可选): 3000 (TCP) 4443 (TCP) 30000-30100 (TCP 和 UDP)

以上可选功能如果不需要, 则无需开放相应端口。

#### 五、 获取 docker 镜像文件

(因 docker 经常升级, 最好先咨询我方, 确定最新版本)

镜像文件下载地址:

<http://ent.allptt.com:59680/TotalkWar/clientDownloadFile.do?filename=docker/release.txt>

或通过邮件或 qq 发送镜像文件 (一个 zip 文件, 约 600MB)。

获取 zip 文件后, 首先解压出 tar 文件, 再在宿主机执行 load, 如:

```
sudo docker load < totalk_v325.tar
```

加载镜像。

以上步骤结束后, 可执行

```
sudo docker images
```

查看是否有对应的镜像信息输出，正常应输入类似如下内容：

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
562245918/totalk	v3.06	c38c70355740	6 weeks ago	1.93GB

**注：本文操作主要针对 Linux 系统。**

对于 Windows 系统，应在命令行环境执行同样命令，去掉开头的 sudo。

Windows 上启动 Docker Desktop 时如果长时间 starting docker engine，可尝试重启电脑，或更换软件版本。参考 <https://zhuanlan.zhihu.com/p/663821762>。

Windows 上启动 docker，不要在软件界面上点击操作，而应在 cmd 命令行里操作。

## 六、 对讲 docker 部署运行

### 1. 创建共享数据卷

因服务器运行时，会在 docker 容器内生成数据库文件、媒体文件等。需在宿主机里创建映射，实时备份，确保容器停止、重启时，数据不丢失。

在宿主机上执行：

```
sudo docker volume create totalk_info
```

再执行：

```
sudo docker volume ls
```

输出应包含下面项：

DRIVER	VOLUME NAME
local	totalk_info

表明创建成功。

宿主机上查看 volume，如：

```
sudo docker volume inspect totalk_info
```

可查看共享卷的宿主机路径，正常是

/var/lib/docker/volumes/totalk\_info/\_data/。在此目录下可看到部分服务程序、库，以及保存数据库文件、媒体文件等的 totalk\_upload 目录。在 docker 运行后，/var/lib/docker/volumes/totalk\_info/\_data/与 docker 容器里的工作目录可以实时同步。

### 2. 运行 docker 镜像

首先请逐字、认真阅读以下注意事项：

●下面命令中尖括号**部分的<image id>**替换为实际 image id，例如 v3.06 的<image id>替换为 c38c70355740。本文档里其他尖括号部分，均用类似方式替换。

●如果要使用网页对讲功能，则需在运行 docker 前先生成 mycert.crt 和 mykey.key 文件，复制到宿主机/var/lib/docker/volumes/totalk\_info/\_data/目录下。文件生成方式是：运行以下命令，并根据实际情况填写信息，生成 mycert.crt 和 mykey.key 文件：

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout ./mykey.key -out mycert.crt
```

使用自行生成的证书的缺点是浏览器打开链接时，会提示风险。正式使用时，建议使用 ZeroSSL 等工具获取正式的证书。

● **-d** 参数运行后直接退出，该容器默认会在后台运行(可通过 `docker ps` 查看是否在运行)。如果去掉 **-d**，则运行后，终端会不断的实时输出 **Log** 信息，无法退出。如果需要监测运行日志可以启动运行该命令。

● 整个命令完整复制粘贴，中间不要加回车！

● 每次 `run` 之前，都要 `docker ps`，确认没有其他对讲服务容器在运行，否则端口冲突。如果有，先 `docker stop <container ID>` 停掉或 `docker rm -f <container ID>` 删掉。

● `-p 443:443` 表示把容器里的 443 端口映射给宿主机的 443 端口。其中第一个 443 表示宿主机里的端口，第二个 443 表示容器的端口。第二个端口不能改变，如果要映射为宿主机的其他端口比如 1443，就改为 `-p 1443:443`。修改其他端口方法类似。

如果要修改 8081 端口，例如改成 8091，则需：1) `docker run` 时“`-p 8091:8081`”；2) 命令行进入 `docker` (具体参考第九部分)，执行

```
vi /usr/local/apache2/htdocs/manage/static/utils/webadmin_config.js
```

把 `server_ip` 行，改成：

```
server_ip: '127.0.0.1:8091', (注意：有的服务器禁止直接访问本机，可设置为 <外网 IP>:8091)
```

保存退出。然后进入 `docker`，重启 `apache2`：

```
/usr/local/apache2/bin/apachectl restart
```

再 `exit` 退出 `docker` 即可。

此时访问 `web` 后台 <http://<ip>:8080/manage>，会看到“服务器 IP”栏，已变成 127.0.0.1:8091，可以正常登录。（如果浏览器显示端口未变，尝试浏览器清缓存或无痕模式）

执行以下命令即可运行 `docker`、启动容器，其中 `<image id>` 替换为真实镜像 `id` (可只输入前 4 位字符)。

```
sudo docker run -d -e TZ=Asia/Shanghai -p 443:443 -p 8080:8080 -p 8081:8081 -p 59638-59639:59638-59639 -p 59638:59638/udp -p 59680:59680 -p 8443:8443 -p 1935:1935 -p 1985:1985 -p 1990:1990 -p 8000:8000/udp -p 10080:10080/udp --restart=always --log-opt max-size=100m --log-opt max-file=3 -v totalk_info:/home/zcx <image id> /bin/bash cmd.sh
```

执行成功，应有类似以下输出：

```
zcx@ubuntu:~$ sudo docker run -d -e TZ=Asia/Shanghai -p 443:443/tcp -p 1935:1935/tcp -p 8080:8080 -p 8081:8081 -p 59638:59638/tcp -p 59638:59638/udp -p 59639:59639/tcp -p 59639:59639/udp -p 59680:59680 -p 8443:8443 -p 1935:1935 -p 1985:1985 -p 1990:1990 -p 8000:8000/udp -p 10080:10080/udp --restart=always --log-opt max-size=100m --log-opt max-file=3 -v totalk_info:/home/zcx/c38c70355740 /bin/bash cmd.sh
970205015b46f43834bc55963da0a764d0058a43fe9ece6d97ddd890ddf491df
```

此时执行 `sudo docker ps`，应能看到刚才运行的容器：

```
zcx@ubuntu:~$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
970205015b46   c38c70355740   "/bin/bash cmd.sh"      6 seconds ago Up 5 seconds  0.0.0.0:443->443/tcp, 0.0.0.0:1935->1935/tcp, 0.0.0.0:8080->8080->8080-8081/tcp, 0.0.0.0:59638->59638-59639/tcp, 0.0.0.0:59680->59680/tcp, 0.0.0.0:59638->59638-59639/udp, 0.0.0.0:59639->59638-59639/tcp, 0.0.0.0:59680->59680/tcp, 0.0.0.0:59638->59639/udp, 0.0.0.0:59639->59638-59639/udp   kind_noether
```

(注：`-p 8081:8081` 也是必须，否则无法登录后台管理网页)

打开浏览器输入 `<宿主机 IP>:8080/manage` 即可访问滔滔后台界面，随后可参考《滔滔对讲 Web 后台使用说明》，进行服务器授权、创建组织、频道、用户等操作。创建好后，客户端侧即可登录、对讲。

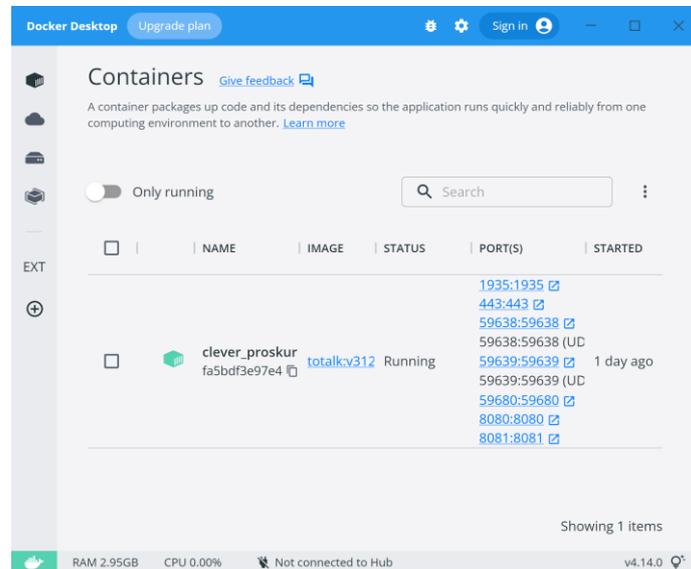
如果打不开后台界面，确认 8080 端口开放，且网络运营商未禁止 8080 端口。也可把 8080 映射为其他端口。另外，可进入 docker，确认 apache2 启动了：

```
ps -ef|grep apache2
```

重新启动 apache2 服务命令：`/usr/local/apache2/bin/apachectl restart`

注意，访问后台管理界面后，一般需先 admin 登录，设置好文件服务器、录音服务器等参数。

在 Windows 平台，也可在 Docker desktop 软件界面查看 docker 状态：



**（可选）docker 里 tomcat 支持 https（docker v329 及以上）：**

对讲系统中，服务器录音、即时通信的文件图片收发、位置轨迹等功能依赖于 tomcat，默认使用 http(59680)端口。在绝大多数情况下，无需额外配置。如果要支持较新 iOS 系统，或强制要求使用 https，则需为 tomcat 服务器配置合法证书。

宿主机上首次运行主服务器 docker 时，宿主机的 `/var/lib/docker/volumes/totalk_info/_data/` 目录下会自动出现 `tomcat.jks` 文件，这个是默认提供的自签名证书。如需支持 https 请求，需自行准备证书并同名替换，然后重新运行 tomcat。同名证书默认需设置密码为 123456。如果需要使用其他密码，则应进入 docker 容器，编辑 `/opt/apache-tomcat-8.5.43/conf/server.xml`，把

```
<Certificate certificateKeystoreFile="/home/zcx/tomcat.jks"
                certificateKeystorePassword="123456"
                certificateKeystoreType="JKS"
            />
```

里的 123456 修改为实际密码。

**（可选）docker 里直播服务支持 https api 接口（docker v330 及以上）：**

如果客户选配了视频直播功能，且需要在网页中集成视频直播的播放功能，且需要使用 https 接口（默认支持 http），则需为直播服务配置合法证书。

宿主机上首次运行主服务器 docker 时，宿主机的 `/var/lib/docker/volumes/totalk_info/_data/` 目录下会自动出现 `streaming.key` 和 `streaming.crt` 文件，这个是默认提供的自签名证书。如需支持 https api，需自行准备证书并同名替换，然后重新运行 docker。

## 七、 对讲服务 docker 升级

### 1. 首先宿主机执行

```
sudo docker ps
```

拿到旧版容器名或容器 ID，如

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
bc4781b4ee2d	c38c70355740	"/bin/bash cmd.sh"	6 days ago	Up 2 hours	0.0.0.0:443->443/tcp, :::443->443/tcp, 0.0.0.0:1935->1935/tcp, :::1935->1935/tcp, 0.0.0.0:8080-8081->8080-8081/tcp, :::8080-8081->8080-8081/tcp, 0.0.0.0:59638-59639->59638-59639/tcp, :::59638-59639->59638-59639/tcp, 0.0.0.0:59680->59680/tcp, :::59680->59680/tcp, 0.0.0.0:59638-59639->59638-59639/udp, :::59638-59639->59638-59639/udp	confident_lichterman

停止旧镜像的容器：

```
sudo docker stop <旧 docker container ID>
```

一般来说，如果之前正确创建和使用了共享数据卷，则 docker 容器的重要数据都会在宿主机保存，并在升级 docker 后自动读取到新容器里。这里稳妥起见，在宿主机查看 `/var/lib/docker/volumes/totalk_info/_data/` 下文件正常存在，特别是 `totalk_upload/totalk.sqlite` 文件确认存在，且最新修改时间正常。

删除旧镜像的容器：（必须执行，以免下次服务器重启时，旧容器自动启动，造成冲突）：

```
sudo docker rm -f <旧 docker container ID>
```

### 2. 升级新版本：

按本文上述方法，获取新版 docker 镜像文件，如 `totalk_v3.07.tar`

执行

```
sudo docker load < totalk_v3.07.tar
```

加载镜像

再按上述运行 docker 的方法，运行新 docker 容器。运行后，可直接浏览器访问<宿主机 IP>:8080/manage 查看授权和用户、频道还在，客户端登录验证。

如果发现数据不在了，就用手动备份的数据恢复：

```
sudo docker cp
```

```
/var/lib/docker/volumes/totalk_info/_data/totalk_upload/totalk.sqlite <新的 container ID>:/home/zcx/totalk_upload/totalk.sqlite
```

此命令将备份好的服务器数据载入新镜像启动的容器。

重启容器进行生效，执行 `docker restart <容器名或容器 ID>`

另外，对于临时测试或个别文件更新，也可不更新整个 docker 版本。例如，只希望更新主程序 `totalkd`，则只需把此文件复制到容器里，替换容器里的 `totalkd`，并重启该程序。这样更新后，即使宿主机重启，也会运行新的程序。

## 八、 客户端软件在线升级

从 docker V331 (2024.01.14) 开始，系统支持客户端在线升级，包括手机 Android app 和 Windows 调度台 (客户端版本也有一定要求)。

例如，Android 客户端现有版本名称是 1.0，版本号是 1。现在要发布新版本 1.1，版本号 2，则需如下操作：

1) 把新版 apk 文件上传到服务器：

稳妥起见，文件名只包含数字或英文字符，如 POC\_V1.1\_2.apk。文件复制到宿主机的 /var/lib/docker/volumes/totalk\_info/\_data/totalk\_upload/client/ 下面。

## 2) 修改配置文件：

vi 打开宿主机的 /var/lib/docker/volumes/totalk\_info/\_data/pttserver.ini，在里面添加或修改以下内容：

```
android_version_code="2"  
android_version_name="V1.1"  
android_file_path="client/POC_V1.1_2.apk"  
android_feature="性能优化, 已知问题解决"
```

对于 windows 调度台软件，使用以下格式：

```
windows_version_code="88"  
windows_version_name="V2.1.9.1"  
windows_file_path="client/Totalk_V2.1.9.exe"  
windows_feature="1.性能优化\n2.已知问题解决"
```

以上引号里的内容，根据实际情况填写。其中 file\_path 后面的 client/ 需要保留。

这样即完成服务器升级配置。此时在手机或 PC 上启动客户端，即会自动检测到新版本，红点提示，并可进一步检查新版本介绍、选择下载升级。

## 九、 音视频通话 docker 部署运行（可选）

对讲系统支持音视频通话，目前支持 Android 和 Windows 客户端。此功能可选，并单独收费。

通话服务器可与主服务器同机部署，也可异机、异地部署。用户量不太大时不妨同机部署，较方便管理。

部署音视频通话，宿主机需额外开通端口：

4443 (TCP)

30000-30100 (TCP 和 UDP)

首先获取音视频通话 docker 镜像 (tar 文件，约 1.5GB，如果是 zip 文件则先解压出 tar 文件)，以 msoup\_v105.tar 为例，部署步骤：

### 1) 加载镜像

```
sudo docker load < msoup_v105.tar
```

### 2) 查看镜像 ID

```
sudo docker images
```

找到 IMAGE ID，如 b5507cb57cfe

### 3) 运行容器

```
sudo docker run -d -e TZ=Asia/Shanghai -p 4443:4443 -p 30000-30100:30000-30100  
-p 30000-30100:30000-30100/udp --restart=always --log-opt max-size=100m --log-opt  
max-file=3 <image ID> /bin/bash cmd.sh <宿主机对外 IP, 注意是 IP 不能是域名>
```

例如 app 登录 IP 是 192.168.1.100，则运行命令是：

```
sudo docker run -d -e TZ=Asia/Shanghai -p 4443:4443 -p 30000-30100:30000-30100  
-p 30000-30100:30000-30100/udp --restart=always --log-opt max-size=100m  
--log-opt max-file=3 b5507cb57cfe /bin/bash cmd.sh 192.168.1.100
```

如果宿主机 IP 地址变化，需先 stop 掉容器，然后重新执行 run 命令启动。  
运行后，在客户端所在网络环境（公网或局域网）下，执行  
curl <IP>:4443  
应有响应，类似 curl: (52) Empty reply from server

#### 4) 配置通话服务器、开通授权

admin 账号登录后台，在“系统状态”栏，正确设置通话服务器地址，一般形如<IP 地址>:4443，例如:192.168.123.1:4443。

另外，在系统状态栏，查看“总视频通话容量”，向服务提供方提供服务器硬件序号，并获取授权码。获取后，可把通话容量分配到要使用的组织里。之后，该组织用户登录，即可使用通话功能。

## 十、 地图服务 docker 部署运行（开发中、可选）

对讲系统支持独立地图服务部署，预计 2023.7 月份发布。目前支持 Android、iOS 和 Windows 客户端。此功能可选，并单独收费。

地图服务可与主服务器同机部署，也可异机、异地部署。用户量不太大时不妨同机部署，较方便管理。

部署地图服务，宿主机需额外开通端口：  
9080 和 9443 (TCP)

首先获取地图服务 docker 镜像(zip 文件，约 200MB)，以 ttmap\_v100.zip 为例，部署步骤：

### 1. 创建共享数据卷

地图服务运行时，需要地图数据、接口程序、https 证书文件等资源。为方便客户管理和更新资源文件，需在宿主机上创建共享数据卷。

在宿主机上执行：

```
sudo docker volume create map_info
```

再执行：

```
sudo docker volume ls
```

输出应包含下面项：

DRIVER	VOLUME NAME
local	map_info

表明创建成功。

宿主机上查看 volume，如：

```
sudo docker volume inspect map_info
```

可查看共享卷的宿主机路径，正常是

/var/lib/containers/storage/volumes/map\_info/\_data。在运行 docker 运行后，此目录与 docker 容器里的工作目录可以实时同步。宿主机上首次运行地图 docker 时，本目录下会自动出现 certs 和 map 两个目录。其中 certs 目录里包含了 3 个 https 要用到的证书等文件，客户如需支持 https 请求，需自行准备证书并同名替换，然后重新运行 docker。

map 目录下部署了地图服务接口，以及简略地图数据。如需添加更多地图数据，或更新地图接口，只需把相应文件复制或覆盖到对应位置即可。

## 2. 运行地图 docker

### 1) 加载镜像

```
sudo docker load < ttmap_v100.tar
```

### 2) 查看镜像 ID

```
sudo docker images
```

找到 IMAGE ID, 如 b5507cb57cfe

### 3) 运行容器

```
sudo docker run -d -e TZ=Asia/Shanghai -p 9080:9080 -p 9443:9443
--restart=always --log-opt max-size=100m --log-opt max-file=3 -v
map_info:/var/www/html <image ID> /bin/bash cmd.sh
```

### 4) 配置地图服务器、授权后启用地图服务

admin 账号登录后台, 在“系统状态”栏, 正确设置地图服务器地址, 一般形如 http(s)://<域名或 IP>:9443, 例如: http://192.168.123.1:9080, 或 https://map.my.com:9443。

另外, 在系统状态栏, 查看“启用地图服务”, 向服务提供方提供服务器硬件序号, 并获取授权码, 即可使用地图功能。

## 十一、常用 docker 操作

如需进入 container, 首先在宿主机执行

```
sudo docker ps
```

查看最后的 container ID, 正常输出:

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
bc4781b4ee2d	c38c70355740	"/bin/bash cmd.sh"	6 days ago	Up 2 hours	0.0.0.0:443->443/tcp, :::443->443/tcp, 0.0.0.0:1935->1935/tcp, :::1935->1935/tcp, 0.0.0.0:8080-8081->8080-8081/tcp, :::8080-8081->8080-8081/tcp, 0.0.0.0:59638-59639->59638-59639/tcp, :::59638-59639->59638-59639/tcp, 0.0.0.0:59680->59680/tcp, :::59680->59680/tcp, 0.0.0.0:59638-59639->59638-59639/udp, :::59638-59639->59638-59639/udp	confident_lichterman

可看到其 container ID 是 bc4781b4ee2d

宿主机执行 `sudo docker exec -it bc4781b4ee2d bash`

即可进入 bash 控制 container

(小技巧: 如果容器运行后立即退出, 可以

```
sudo docker run -itd d2e881e27fcf /bin/bash -c "nohup ping -i 1000
```

[www.baidu.com](http://www.baidu.com)”保持其不退出。这样可以 `docker exec` 进入容器进行修改、配置等)

停止 docker:

```
sudo stop <container ID>
```

强行停止一个状态是 Restarting 或 Exited 的 docker:

```
sudo docker rm -f <container id>
```

批量删除所有状态为 exited 的容器:

```
sudo docker rm $(sudo docker ps -qf status=exited)
```

批量删除所有状态为 created 的容器:

```
sudo docker rm $(sudo docker ps -qf status=created)
```

查看 docker 里服务器输出的日志（最近 xx 行）：

```
docker logs --tail 100 <container id>
```

实时显示日志：

```
docker logs -f <container id>
```

清理 docker 的日志：

```
truncate -s 0 /var/lib/docker/containers/*/*-json.log
```