

docker 安装 solo 及相关配置

作者: [JayGao](#)

原文链接: <https://ld246.com/article/1650443620382>

来源网站: [链滴](#)

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

<p></p>
<h2 id="不使用默认网卡-创建新网卡">不使用默认网卡，创建新网卡</h2>
<p>-- 组建 docker 内内部网络

docker network create -d bridge intranet</p>
<p>-- 查看 docker 网络列表（网卡）

docker network ls</p>
<p>参考文章：Docker 容器的常用网络模式与容器间互访 - 知乎 (zhihu.com)</p>
<p>参考文章：Docker 容器互访三种式 - 三只松鼠 - 博客园 (cnblogs.com)</p>
<p>参考文章：Docker 容器连接 | 菜鸟教程 (runoob.com)</p>
<h2 id="安装mysql">安装 mysql</h2>
<p>-- 创建 mysql 容器，不外挂文件

docker run -itd --name mysql --restart always -p 3306:3306 -e TZ=Asia/Shanghai -e MYSQL_ROOT_PASSWORD=123456 --network intranet mysql:latest --character-set-server=utf8mb4 --collation-server=utf8mb4_0900_ai_ci</p>
<p>--创建 mysql 容器，外挂数据文件、配置文件</p>
<p>docker run -itd --name mysql --restart always -p 3306:3306 -e TZ=Asia/Shanghai -e MYSQL_ROOT_PASSWORD=123456 -v /home/mysql/config:/etc/mysql/conf.d -v /home/mysql/data:/var/lib/mysql mysql:latest --character-set-server=utf8mb4 --collation-server=utf8mb4_0900_ai_ci</p>
<p>**备注：通用 mysql log 日志文件默认没有开启（off 状态），所以不需要外挂，若需要外挂则需要在配置文件中开启日志记录功能，并指定日志文件路径，由于日志文件时间长了会大，不建议开启。</p>
<p>查看日志文件状态核路径：</p>
<p>SHOW VARIABLES LIKE '%general%';</p>
<table>
<thead>
<tr>
<th>序号</th>
<th>变量名</th>
<th>值</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>general_log</td>
<td>OFF</td>
</tr>
<tr>
<td>2</td>
<td>general_log_file</td>
<td>/var/lib/mysql/aafe7bdcdebe.log</td>
</tr>
</tbody>
</table>
<h2 id="安装solo">安装 solo</h2>
<p>-- cnd 加速 加速不咋样，很慢


```

docker run -itd --name solo -p 443:443 -p 80:80<br>
--env TZ=Asia/Shanghai<br>
--env RUNTIME_DB="MYSQL"<br>
--env JDBC_USERNAME="root"<br>
--env JDBC_PASSWORD="123456"<br>
--env JDBC_DRIVER="com.mysql.cj.jdbc.Driver"<br>
--env JDBC_URL="jdbc:solo?allowPublicKeyRetrieval=true&allowMultiQueries=true&
p:autoReconnect=true<br>
&failOverReadOnly=false&useUnicode=true&characterEncoding=utf8&u
eSSL=false&serverTimezone=Asia/Shanghai"<br>
--restart always<br>
--network intranet<br>
b3log/solo<br>
--listen_port=80<br>
--server_scheme=http<br>
--server_host=domain.com<br>
--server_port=</p>
<p>###后面这 4 行是 cdn 资源加速配置<br>
--static_server_scheme=https<br>
--static_server_host=cdn.jsdelivr.net<br>
--static_server_port=<br>
--static_path=/gh/88250/solo/src/main/resources</p>
<p><strong>-- 不用 cdn</strong><br>
docker run -itd --name solo -p 443:443 -p 80:80<br>
--env TZ=Asia/Shanghai<br>
--env RUNTIME_DB="MYSQL"<br>
--env JDBC_USERNAME="root"<br>
--env JDBC_PASSWORD="123456"<br>
--env JDBC_DRIVER="com.mysql.cj.jdbc.Driver"<br>
--env JDBC_URL="jdbc:solo?allowPublicKeyRetrieval=true&allowMultiQueries=true&
p:autoReconnect=true<br>
&failOverReadOnly=false&useUnicode=true&characterEncoding=utf8&u
eSSL=false&serverTimezone=Asia/Shanghai"<br>
--restart always<br>
--network intranet<br>
b3log/solo<br>
--listen_port=80<br>
--server_scheme=http 协议/https 协议<br>
--server_host=domain.com<br>
--server_port=外部访问的端口没有默认 80</p>
<p>参考文章: <a href="https://ld246.com/article/1492881378588#%E8%8E%B7%E5%8F%9
%E6%9C%80%E6%96%B0%E9%95%9C%E5%83%8F">Solo 用户指南 - 链滴 (ld246.com)</a><
p>
<h2 id="安装gitlab">安装 gitlab</h2>
<p>--shm-size 虚拟内存大小</p>
<p>docker run -itd --hostname gitlab.domain.com -p 443:443 -p 80:80 -p 22:22<br>
-e TZ=Asia/Shanghai<br>
--name gitlab<br>
--restart always<br>
-v /srv/gitlab/config:/etc/gitlab<br>
-v /srv/gitlab/logs:/var/log/gitlab<br>
-v /srv/gitlab/data:/var/opt/gitlab<br>
--shm-size 256m<br>
--network intranet<br>

```

```
gitlab/gitlab-ce:latest</p>
<p>-- 这个目录是 gitlab 安装部署相关目录，可以修改静态页面：例如增加备案号（icp 和公安） </p>
<p>-v /srv/gitlab/install:/opt/gitlab 这里增加安装目录</p>
<h2 id="安装nginx">安装 nginx</h2>
<p>docker run -itd --name=nginx -p 80:80 -p 443:443 --restart=always<br>
-v /srv/nginx/conf/nginx.conf:/etc/nginx/nginx.conf<br>
-v /srv/nginx/conf.d:/etc/nginx/conf.d/<br>
-v /srv/nginx/cert:/etc/nginx/cert/<br>
-v /srv/nginx/logs:/var/log/nginx/<br>
-v /srv/nginx/html:/usr/share/nginx/html<br>
-e TZ=Asia/Shanghai<br>
--network intranet<br>
nginx:1.21.6</p>
<h2 id="安装pgsql">安装 pgsql</h2>
<p>docker pull postgres:14.2<br>
docker run -itd --name pgsql --restart always -e POSTGRES_PASSWORD=123456 -p 5432:5432
--network intranet postgres:14.2</p>
<h2 id="安装mariadb">安装 mariadb</h2>
<p>docker pull mariadb:10.6<br>
docker run -itd --name mariadb -p 3306:3306 --restart always --env MARIADB_USER=root --
nv MARIADB_PASSWORD=123456 --env MARIADB_ROOT_PASSWORD=123456 --network int
anet mariadb:10.6</p>
<h2 id="安装GitHub下载加速服务">安装 GitHub 下载加速服务</h2>
<p>docker pull hunsh/gh-proxy-py:latest</p>
<p>docker run -itd --name="gh-proxy-py" -p 0.0.0.0:80:80 --restart=always hunsh/gh-pr
xy-py:latest</p>
```