



文档工具套件部署手册

友虹（北京）科技有限公司

2023年8月21日



修改记录

时间	章节	修改类型	描述	作者	版本
2023/08/21	全部	A	撰写部署 1.4 版本	李伟	1.4

*修改类型分为 **A** - 新增 **M** - 修改 **D** - 删除



目录

1. 文档概述	1
2. 服务器资源	2
3. 部署步骤	2
3.1. 下载文件	2
3.1.1. 下载	2
3.1.2. 解压系统文件	2
3.1.3. 修改 nginx 配置	3
3.2. 安装 docker	3
3.2.1. 下载 docker	3
3.2.2. 安装 docker	3
3.2.3. 配置 docker 为系统服务输入	3
3.2.4. 添加文件权限并启动 docker，执行如下命令	5
3.2.5. 验证 docker 安装是否成功	5
3.3. 安装 docker-compose	5
3.3.1. docker-compose 下载	5
3.3.2. 配置 docker-compose	5
3.3.3. 验证	6
3.4. 将下载的镜像导入 docker	6
3.5. 启动 nacos 镜像并配置	6
3.5.1. 启动 nacos 镜像	6
3.5.2. 登录 nacos 控制台	7
3.5.3. 新建命名空间	7
3.6. 导入配置	8
3.6.1. 导入配置压缩包	8
3.6.2. 修改配置	9
3.7. 更改 nacos 密码	9
3.8. 退出	10
3.9. 执行 docker-compose	10
3.9.1. 更改配置	10
3.9.2. 运行	11
3.10. 安装完成检测	12
4. 系统管理运维	13
4.1. 容器的启停	13
5. 技术支持	14



1. 文档概述

《文档工具套件部署手册》主要内容是提供友虹文档工具套件部署操作的用户文档，用户单位按照部署手册进行对文档工具套件系统进行安装与部署，并且文档中提供文档工具套件的相关运维内容，便于系统运维人员进行系统维护。

2. 服务器资源

为了保证文档工具套件系统的性能与稳定性，我们为此提供了最优的系统部署所需服务器资源与部署节点优化配置方案，请按照以下列表的列表准备相应的服务器资源。

服务器配置：

类型	CPU	内存	存储	系统
最低配置	8 核	32G	1T	CentOS7 x64
推荐配置	16 核	32G	1T	CentOS7 x64

上面为服务器部署的资源配置，后续可根据系统的运行与使用情况，相应的扩容或缩减相关资源。

3. 部署步骤

3.1. 下载文件

3.1.1. 下载

<http://files.scfile.cn/tool/aarch64-1.5/>

将地址下的所有文件下载至/home 目录下

3.1.2. 解压系统文件

```
解压 youhong.zip
```

```
cd /home
```

```
unzip youhong.zip
```

最终形成目录结构为/home/youhong/tool/.....

3.1.3. 修改 nginx 配置

将配置文件第三行的 IP 地址改为本机真实 IP

```
vi /home/youhong/tool/nginx/conf.d/liankong-3.0.conf
```

```
1 server {
2     listen 80;
3     server_name 192.168.1.217;
4     add_header Access-Control-Allow-Origin *;
5     add_header Access-Control-Allow-Credentials 1
```



3.2. 安装 docker

3.2.1. 下载 docker

```
wget -c https://download.docker.com/linux/static/stable/aarch64/docker-19.03.6.tgz
```

3.2.2. 安装 docker

```
tar -xvf docker-19.03.06.tgz
sudo mv docker/* /usr/bin/
```

3.2.3. 配置 docker 为系统服务输入

```
sudo vim /etc/systemd/system/docker.service

[Unit]

Description=Docker Application Container Engine

Documentation=https://docs.docker.com

After=network-online.target firewalld.service

Wants=network-online.target

[Service]

Type=notify
```



```
# the default is not to use systemd for cgroups because the delegate issues still
# exists and systemd currently does not support the cgroup feature set required
# for containers run by docker

ExecStart=/usr/bin/dockerd

ExecReload=/bin/kill -s HUP $MAINPID

# Having non-zero Limit*s causes performance problems due to accounting overhead
# in the kernel. We recommend using cgroups to do container-local accounting.

LimitNOFILE=infinity

LimitNPROC=infinity

LimitCORE=infinity

# Uncomment TasksMax if your systemd version supports it.
# Only systemd 226 and above support this version.

#TasksMax=infinity

TimeoutStartSec=0

# set delegate yes so that systemd does not reset the cgroups of docker containers

Delegate=yes

# kill only the docker process, not all processes in the cgroup

KillMode=process

# restart the docker process if it exits prematurely

Restart=on-failure

StartLimitBurst=3

StartLimitInterval=60s
```

```
[Install]
```

```
WantedBy=multi-user.target
```

3.2.4. 添加文件权限并启动 docker，执行如下命令

```
chmod +x /etc/systemd/system/docker.service    添加文件权限
```

```
systemctl daemon-reload    重载 unit 配置文件
```

```
systemctl start docker    启动 Docker
```

```
systemctl enable docker.service    设置开机自启
```

3.2.5. 验证 docker 安装是否成功

```
systemctl status docker    查看 Docker 状态
```

```
docker -v    查看 Docker 版本
```

3.3. 安装 docker-compose

3.3.1. docker-compose 下载

```
wget
```

```
https://github.com/docker/compose/releases/download/v2.18.1/docker-compose-linux-aarch64
```

3.3.2. 配置 docker-compose

```
#移动
```

```
mv docker-compose-linux-aarch64 /usr/local/bin/docker-compose
```

```
#授权文件权限
```

```
sudo chmod +x /usr/local/bin/docker-compose
```

3.3.3. 验证

```
docker-compose -v
```

3.4. 将下载的镜像导入 docker

依次执行如下 shell 命令

```
cd /home
```

```
docker load -i auth-server.tar
```

```
docker load -i dict-server.tar
```

```
docker load -i document-server.tar
```

```
docker load -i gateway-server.tar
```

```
docker load -i nacos.tar
```

```
docker load -i nginx.tar
```

```
docker load -i operation-server.tar
```

```
docker load -i reader-server.tar
```

```
docker load -i redis.tar
```

```
docker load -i tc-server.tar
```

```
docker load -i tool-db.tar
```

```
docker load -i tool-server.tar
```

```
docker load -i user-server.tar
```

```
docker load -i front.tar
```

3.5. 启动 nacos 镜像并配置

3.5.1. 启动 nacos 镜像

运行 shell 命令,需要注意要将命令中的 ip192.168.1.225 替换为本机真实 ip

```
docker run -itd --name nacos-service --net my_network --restart=always
--env PREFER_HOST_MODE=192.168.1.225 --env MODE=standalone --env
NACOS_AUTH_ENABLE=true -p 8848:8848 -p 9848:9848 -p 9849:9849
nacos/nacos-server:v2.1.1-slim
```

```
docker run -itd --name nacos-service --net my_network --restart=always
--env PREFER_HOST_MODE=192.168.1.225 --env MODE=standalone --env
NACOS_AUTH_ENABLE=true -p 8848:8848 -p 9848:9848 -p 9849:9849
nacos/nacos-server:v2.1.1-slim
```

替换为本机真实ip

3.5.2. 登录 nacos 控制台

启动成功后,进入 nacos 控制台进行配置

使用浏览器访问地址 [http://本机 ip:8848/nacos](http://本机ip:8848/nacos)

用户名:nacos

密码:nacos

3.5.3. 新建命名空间

进入命名空间页面



点击右侧的新建命名空间按钮



分别输入 private 然后确定

新建命名空间 ×

命名空间ID(不填则自动生成): 1

* 命名空间名: 2

* 描述: 3

4

3.6. 导入配置

3.6.1. 导入配置压缩包

1. 点击配置管理菜单
2. 点击配置列表菜单
3. 点击 private 命名空间
4. 点击导入配置按钮



点击上传文件后选择下载的 nacos_config_1.4_私有化.zip 压缩包后上传



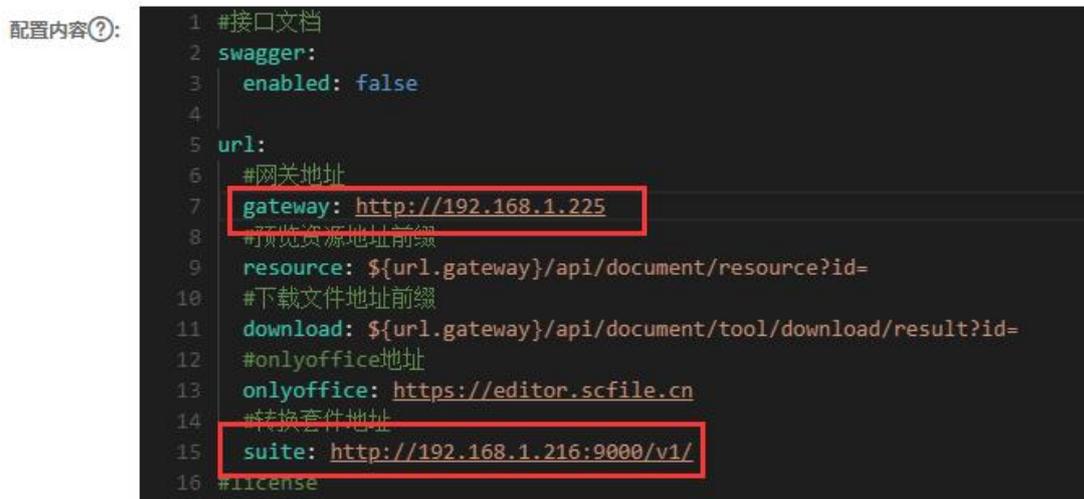
3.6.2. 修改配置

导入成功后,点击第 2 页,找到 public-base.yaml 配置点击编辑



找到第 7 行网关地址,将 ip 改为本机真实 ip

找到第 15 行转换套件地址,将 ip 和端口改为真实的转换套件后台服务的 ip+端口



3.7. 更改 nacos 密码

鼠标移动到右上角 nacos 并悬浮,然后点击修改密码



将密码修改为 youhong



密码重置

* 用户名 nacos

* 密码

* 确认密码

确定 取消

3.8. 退出

设置完毕,退出 nacos



3.9. 执行 docker-compose

3.9.1. 更改配置

编辑 docker-compose.yml 文件
找到倒数第 2 行,将 ip 改为本机真实 ip

```
160 tool-front:
161   image: webtools-shandong:v1.0.0
162   container_name: tool-front
163   networks:
164     - my_network
165   ports:
166     - "3000:3000"
167   environment:
168     - API_BASEURL=http://192.168.1.225/api
169   restart: always
170
```

3.9.2. 运行

执行 shell 脚本,运行 docker-compose

```
docker-compose up -d
```

等待 3-5 分钟,后台服务启动完毕

3.10. 安装完成检测

部署完成之后，使用浏览器输入地址：192.0.0.1，注意这里的 IP 地址是实体物理机的 ip 地址，后续的系统使用详见《文档工具套件管理员用户手册》、《文档工具套件用户手册》

4. 系统管理运维

4.1. 容器的启停

查看所有容器 `docker ps -a`

启动容器 `docker start 容器 id`

停止容器 `docker stop 容器 id`

重启容器 `docker restart 容器 id`



5. 技术支持

以下为文档工具套件技术相关技术人员与联系方式：

姓名：齐文亮，联系方式：15600173212

姓名：李伟 ， 联系方式：15010181162