

按照官网文档部署（我服务器是centos7），参数基本都没问题，只需要自己改部分信息就行，官网文档地址：<https://cws-docs.pages.dev/zh/guide/quick-start.html>

### 1、创建目录

```
cd ~  
mkdir cws && cd cws  
mkdir -p data/config
```

### 2、设定mongodb密码，需要自己修改“密码”

```
# MongoDB 密码  
export MONGODB_PASSWORD=密码  
# 初始管理员密码  
export INITIAL_ADMIN_PASSWORD=密码
```

### 3、创建docker容器

```
docker run -it --rm \  
-v $PWD/data/config:/tmp/config \  
moeakwak/chatgpt-web-share:latest \  
python /app/backend/manage.py create_config -O  
/tmp/config --generate-secrets --mongodb-url  
"mongodb://cws:${MONGODB_PASSWORD}@mongo:27017"  
--initial-admin-password "${INITIAL_ADMIN_PASSWORD}" --  
chatgpt-base-url http://ninja:7999/backend-api/
```

提示没有安装docker，先安装配置docker yum源

```
yum-config-manager --add-repo  
https://download.docker.com/linux/centos/docker-ce.repo
```

再重新安装docker

```
yum install -y docker-ce docker-ce-cli containerd.io docker-  
compose-plugin
```

然后启动docker  
systemctl start docker

设置开机启动docker  
systemctl enable docker

重新执行创建docker 容器命令

#### 4、创建docker compose文件（目前还是在cws目录）

先执行这些命令设置环境变量

```
echo "TZ=Asia/Shanghai" > .env  
echo "MONGO_INITDB_DATABASE=cws" >> .env  
echo "MONGO_INITDB_ROOT_USERNAME=cws" >> .env  
echo  
"MONGO_INITDB_ROOT_PASSWORD=$MONGODB_PASSWORD" >> .env
```

编辑docker-compose文件，将内容粘贴进去（这里ninja端口我没有注释，为了复制到vim编辑器方便，以及后面要手动上传har文件，上传完可以关闭，熟悉的话也可以关闭手动上传）  
vim docker-compose.yml

```
version: "3"
```

```
services:
```

```
  chatgpt-web-share:
```

```
    image: ghcr.io/moeakwak/chatgpt-web-share:latest
```

```
    container_name: cws
```

restart: unless-stopped

ports:

- 5000:80

volumes:

- ./data:/app/backend/data

environment:

- TZ=\${TZ}
- CWS\_CONFIG\_DIR=/app/backend/data/config

depends\_on:

- mongo

mongo:

container\_name: mongo

image: mongo:6.0

restart: always

ports:

- 27017:27017

volumes:

- ./mongo\_data:/data/db

environment:

MONGO\_INITDB\_DATABASE:

\${MONGO\_INITDB\_DATABASE}

MONGO\_INITDB\_ROOT\_USERNAME:

\${MONGO\_INITDB\_ROOT\_USERNAME}

MONGO\_INITDB\_ROOT\_PASSWORD:

\${MONGO\_INITDB\_ROOT\_PASSWORD}

ninja:

image: ghcr.io/gngpp/ninja:latest

container\_name: ninja

restart: unless-stopped

command: run --disable-webui

ports:

- "7999:7999"

environment:

- TZ=Asia/Shanghai

volumes:

- "./har/gpt3:/root/.gpt3"
- "./har/gpt4:/root/.gpt4"

保存后执行容器启动，先使用前台运行方式查看启动有没有报错  
docker compose up

没有报错会显示如下信息：

```
cws | 2023/11/25 19:41:10.054 INFO:
[cws.api.database.mongodb] MongoDB initialized. dataSize:
0.00 MB, objects: 2
cws | 2023/11/25 19:41:10.338 INFO: [cws.main] Using
http://ninja:7999/backend-api/ as ChatGPT base url
cws | 2023/11/25 19:41:10.339 INFO: [cws.main] Sync
conversations on startup disabled.
cws | 2023/11/25 19:41:10.339 INFO: [uvicorn.error]
Application startup complete.
cws | 2023/11/25 19:41:10.339 INFO: [uvicorn.error]
Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to
quit)
```

使用网页打开IP：端口测试（服务器:5000端口）

如果网页可以打开，那回到服务器ctrl+c关闭

5、登陆chatgpt官网，再打开这个链接获取token信息，复制token信息

<https://chat.openai.com/>

<https://chat.openai.com/api/auth/session>

将token信息写入文件（当前在cws目录下）： vim  
data/config/credentials.yaml  
openai\_web\_access\_token: (你的token, 冒号后面留个英文空格, 复制token进去, 不要双引号)  
openai\_api\_key: (这个不管)

例如（实际更长，这里截取了，3.5的token也支持，不过只能使用3.5模型，用4就会报错）：  
openai\_web\_access\_token:  
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6Ikp1UaEV  
openai\_api\_key:

重新后台启动

docker compose up -d

6、上传har文件（不上传使用4问答时会报错，har文件获取方式：打开官网gpt对话框，按f12然后点network，问一个问题，回到开发工具看到network里面有一个35536E1E-65B4-4D96-9D97-6ADB7EFF8147，右键这个保存为har文件）：

获取回复失败

OpenAI 服务器错误 (500 错误)，可能是 OpenAI 服务故障或者模型接口发生变更，请告知管理员

文件上传地址：服务器ip:7999/har/upload  
上传时选择gpt4：

## Upload a .har file extension



Type

GPT-4



Uploads

Only files in .har format are allowed to be uploaded.

FileName	Actions
chat.openai.com.har	 

刷新页面再问gpt4，回复正常（docker compose down关闭服务，去把ninja的端口屏蔽掉防止别人乱上传，docker compose up -d重启服务）

备注：如果已在其他服务器部署过，要将账号信息迁移到当前服务器，把cws/data下的databas.db拷贝过来覆盖就好，覆盖过程要关闭服务