ShopHeatServer 使用说明

- ShopHeatServer 使用说明
 - 。 系统要求
 - 配置 Docker 环境
 - 获取 ShopHeatServer:1.8 docker 镜像
 - o 运行 docker 镜像
 - 。 访问 Web 端界面
 - 安装Docker CE
 - 安装Nvidia Docker
 - o Docker 运行测试
 - 启动 Docker CE
 - 建立docker用户组
 - 测试 Docker 是否安装成功

系统要求

- Ubuntu 18.04
- Nvidia Driver

配置 Docker 环境

需要确保

nvidia-docker

可以正常运行,具体安装步骤见下文

获取 ShopHeatServer:1.8 docker 镜像

docker pull 117.71.55.48:5000/shopheatserver:1.8

运行 docker 镜像

nvidia-docker run -p 4444:4000 -itd shopheatserver:1.8 /bin/bash

访问 Web 端界面

<Ubuntu-ip-address> 为运行 docker 的服务器 ip 地址

<Ubuntu-ip-address>:4444

安装Docker CE

由于apt源使用HTTPS以确保软件下载过程中不被篡改。因此,我们首先需要添加HTTPS传输的软件包以及CA证书。

```
$ sudo apt-get update

$ sudo apt-get install \
    apt-transport-https \
    ca-certificates \
    curl \
    software-properties-common
```

鉴于国内网络问题,建议使用国内源,官方源在注释中查看。

为了确认所下载软件包的合法性,需要添加软件源的GPG密钥。

```
$ curl -fsSL https://mirrors.ustc.edu.cn/docker-ce/linux/ubuntu/gpg | sudo apt-key add -
# 官方源
# $ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

然后需要向source.list中添加Docker软件源

```
$ sudo add-apt-repository \
    "deb [arch=amd64] https://mirrors.ustc.edu.cn/docker-ce/linux/ubuntu \
    $(lsb_release -cs) \
    stable"

# 官方源
#$ sudo add-apt-repository \
    "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
    $(lsb_release -cs) \
    stable"
```

安装apt软件包缓存,并安装docker-ce:

```
$ sudo apt-get update
$ sudo apt-get install docker-ce
```

安装Nvidia Docker

同理,安装 Nvidia Docker:

```
$ curl -s -L https://nvidia.github.io/nvidia-docker/gpgkey | sudo apt-key add -
$ distribution=$(. /etc/os-release;echo $ID$VERSION_ID)
$ curl -s -L https://nvidia.github.io/nvidia-docker/ubuntu16.04/nvidia-docker.list
| sudo tee /etc/apt/sources.list.d/nvidia-docker.list
$ sudo apt-get update
$ sudo apt-get install -y nvidia-docker2
$ sudo pkill -SIGHUP dockerd
```

deb https://nvidia.github.io/libnvidia-container/ubuntu18.04/amd64 / deb https://nvidia.github.io/nvidia-container-runtime/ubuntu18.04/amd64/ deb https://nvidia.github.io/nvidia-docker/ubuntu18.04/amd64/ https://nvidia.github.io/nvidia-container-runtime/ubuntu18.04/amd64/InRelease

Docker 运行测试

启动 Docker CE

运行docker:

```
$ sudo systemctl enable docker #加入开机启动项
$ sudo systemctl start docker #运行docker
# $ sudo systemctl status docker #查看状态
```

运行Nvidia-docker:

```
# $ sudo systemctl enable nvidia-docker #加入开机启动项
# $ sudo systemctl start nvidia-docker #运行docker
# $ sudo systemctl status nvidia-docker #查看状态
```

建立docker用户组

默认情况下,docker命令会使用 Unix socket 与 Docker 引擎通讯。而只有root用户和docker 组的用户才可以访问 Docker 引擎的 Unix socket。出于安全考虑,最好将当前用户加入到 docker用户组中。

```
$ sudo groupadd docker
$ sudo gpasswd -a $USER docker
$ newgrp docker
```

测试 Docker 是否安装成功

测试docker:

\$ docker run hello-world

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

- 1. The Docker client contacted the Docker daemon.
- 2. The Docker daemon pulled the "hello-world" image from the Docker Hub. (amd64)
- 3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
- 4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

\$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID: https://hub.docker.com/

For more examples and ideas, visit: https://docs.docker.com/get-started/

若能正常输出以上信息,说明docker安装成功。

测试nvidia-docker:

			Drive	er Version: 4	30.50	CUDA Versio	on: 10.1
GPU Fan	Name Temp	Perf	Persistence Pwr:Usage/Ca	-M Bus-Id ap M	Disp.A Disp.A	Volatile GPU-Util	Uncorr. ECC Compute M.
0 27%	GeFor	ce RTX P8	608 Off 1W / 950W	00000000: 77MiB		0%	N/A Default
1 27%	GeFor	ce RTX P8	608 Off 21W / 950W	00000000: 1MiB	03:00.0 Off 3 / 99999MiB	0%	N/A Default
				·			+
	esses:		Type Proce	ess name			GPU Memory Usage