

第 2 节

Docker Desktop for Windows安装

本课内容

- 演示本地安装Docker Desktop for Windows专业版
- 演示启用Docker Desktop中的K8s



系统需求

The screenshot shows the Docker Desktop for Windows installation page on the Docker documentation website. The page has a blue header with the Docker logo and navigation links like 'Search the docs', 'Guides', 'Product manuals', 'Glossary', 'Reference', and 'Samples'. A sidebar on the left lists various Docker components with 'Docker Desktop for Windows' expanded, showing 'Getting started', 'Install Docker Desktop for Windows' (which is selected), 'Dashboard', and 'Deploy on Kubernetes'. The main content area has a title 'Install Docker Desktop on Windows' with an estimated reading time of 6 minutes. It states that Docker Desktop for Windows is the Community version and can be downloaded from Docker Hub. A prominent red box highlights the 'Download from Docker Hub' button. Below it, a note says that by downloading, you agree to the Docker Software End User License Agreement and the Docker Data Processing Agreement. Another red box highlights the 'System Requirements' section, which lists the following:

- Windows 10 64-bit: Pro, Enterprise, or Education (Build 15063 or later).
- Hyper-V and Containers Windows features must be enabled.
- The following hardware prerequisites are required to successfully run Client Hyper-V on Windows 10:
 - 64 bit processor with [Second Level Address Translation \(SLAT\)](#)
 - 4GB system RAM
 - BIOS-level hardware virtualization support must be enabled in the BIOS settings. For more information, see [Virtualization](#).

A note states: 'Note: Docker supports Docker Desktop on Windows based on Microsoft's support lifecycle for Windows 10 operating system. For more information, see the [Windows lifecycle fact sheet](#)'.

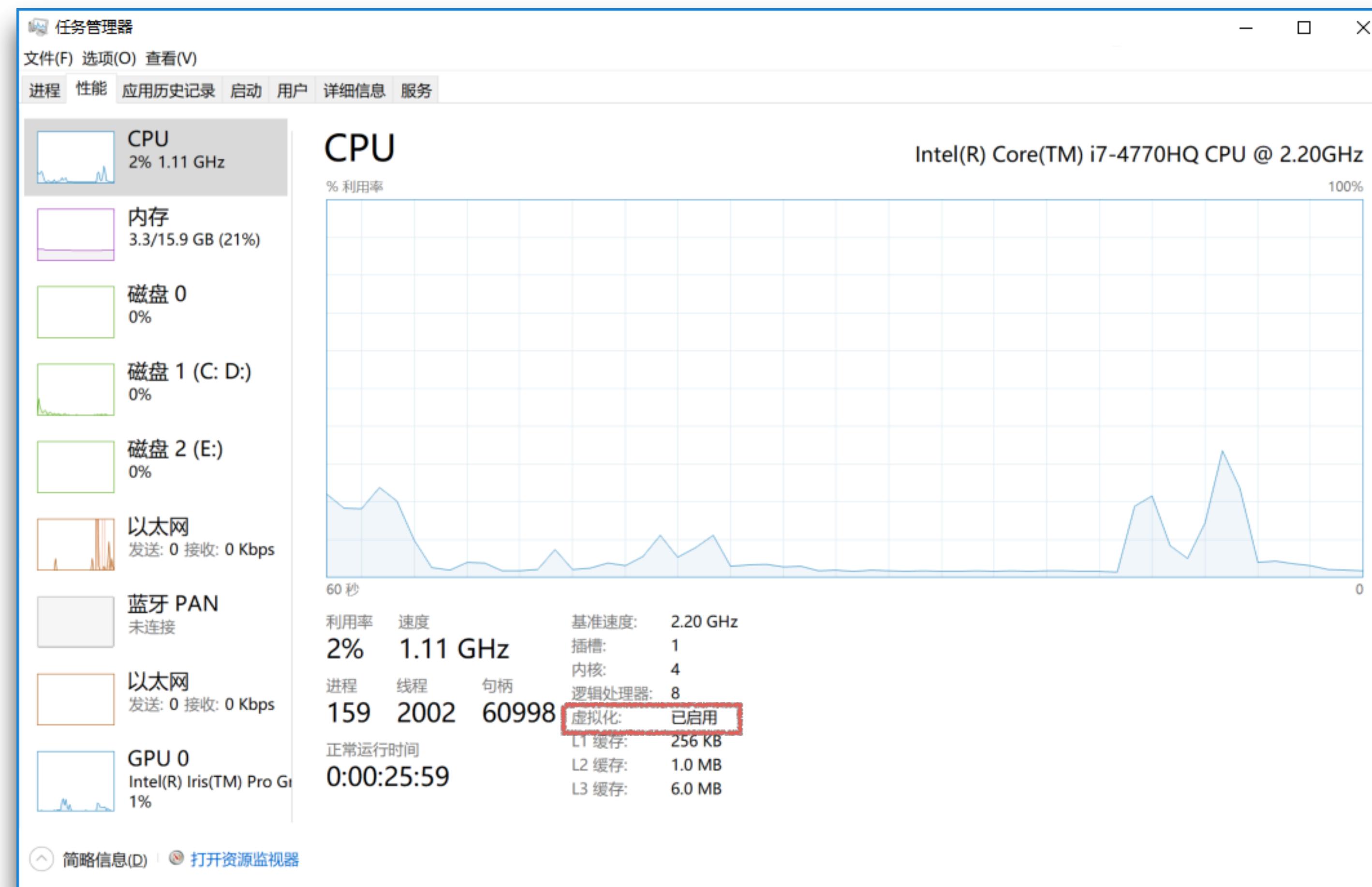
At the bottom, there is a note about Docker Toolbox and Docker Machine users regarding Microsoft Hyper-V requirements. A 'Get Docker' button is located at the bottom left.

<https://docs.docker.com/docker-for-windows/install/>

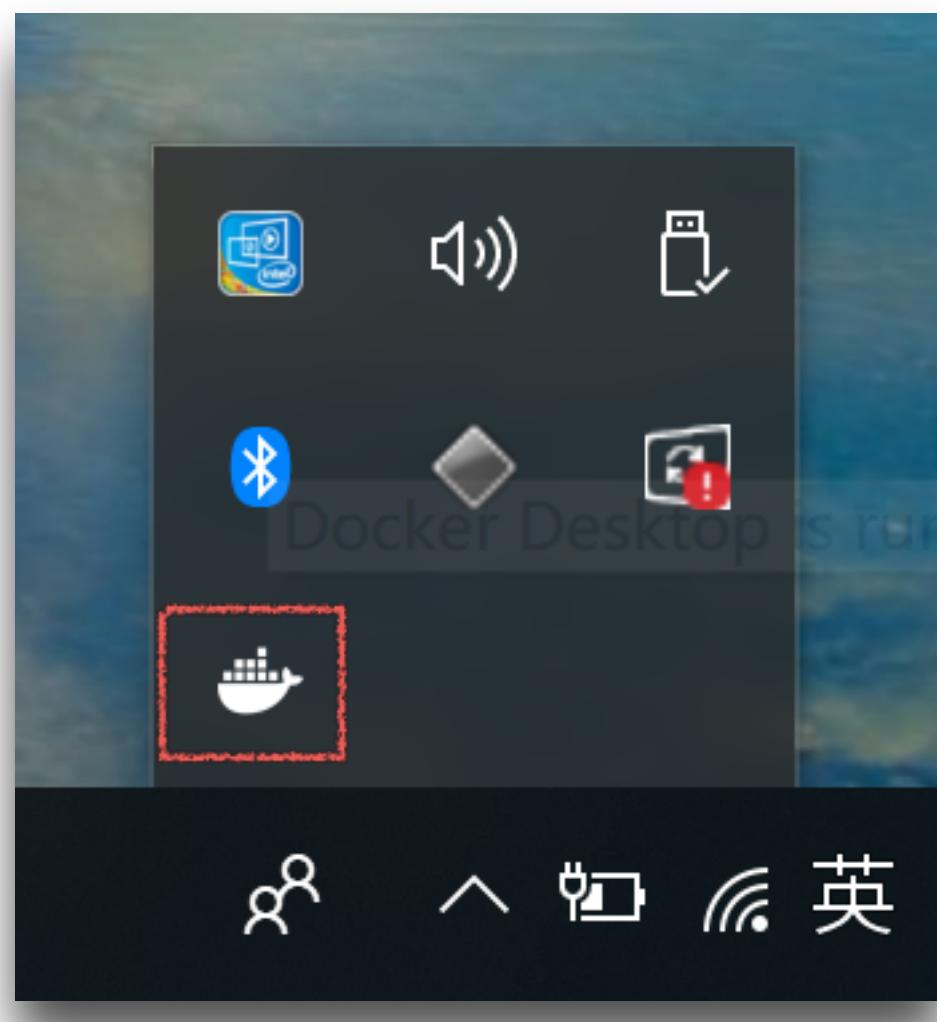
Windows专业版



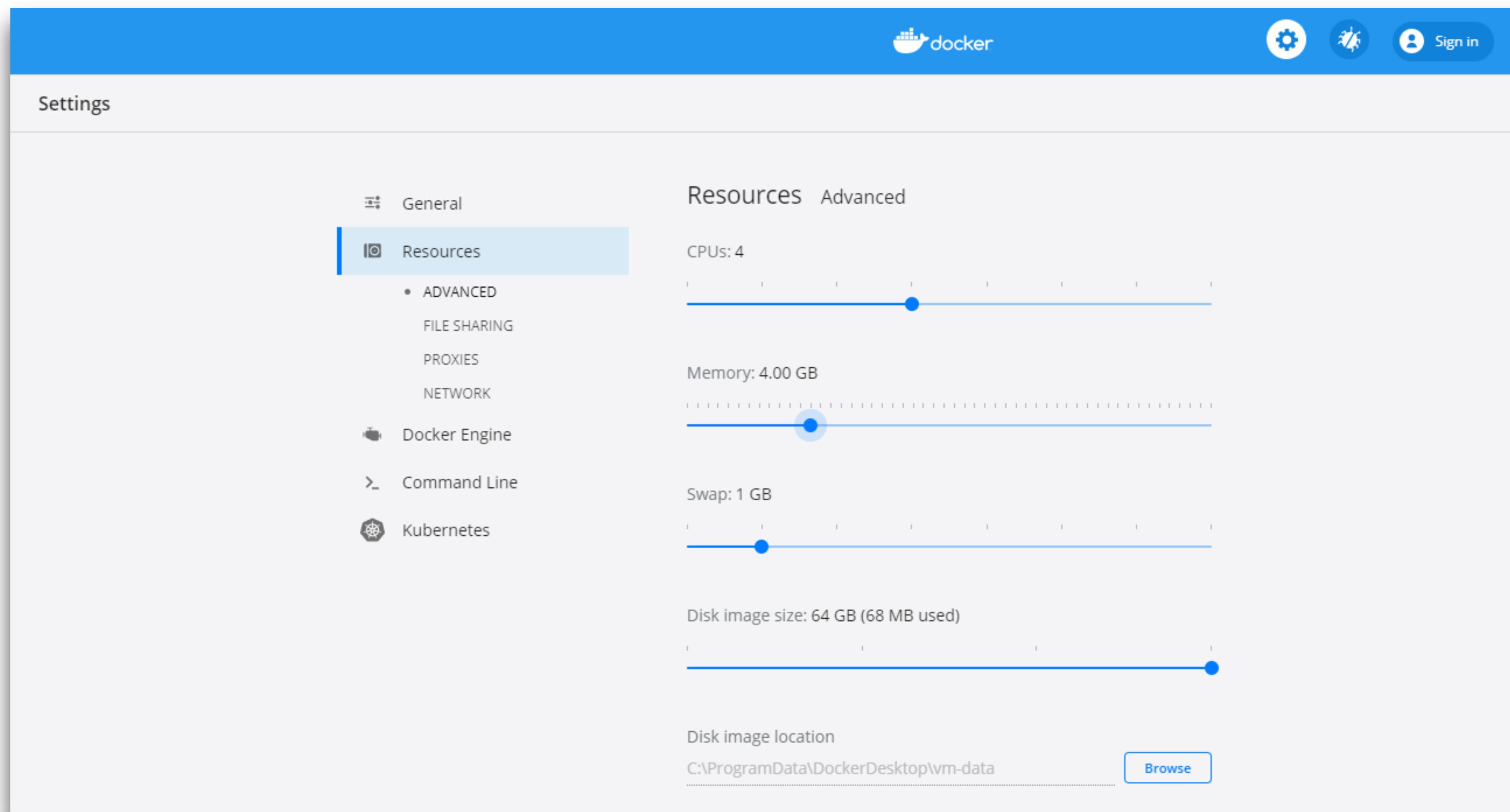
启用虚拟化



成功安装



容器资源设置

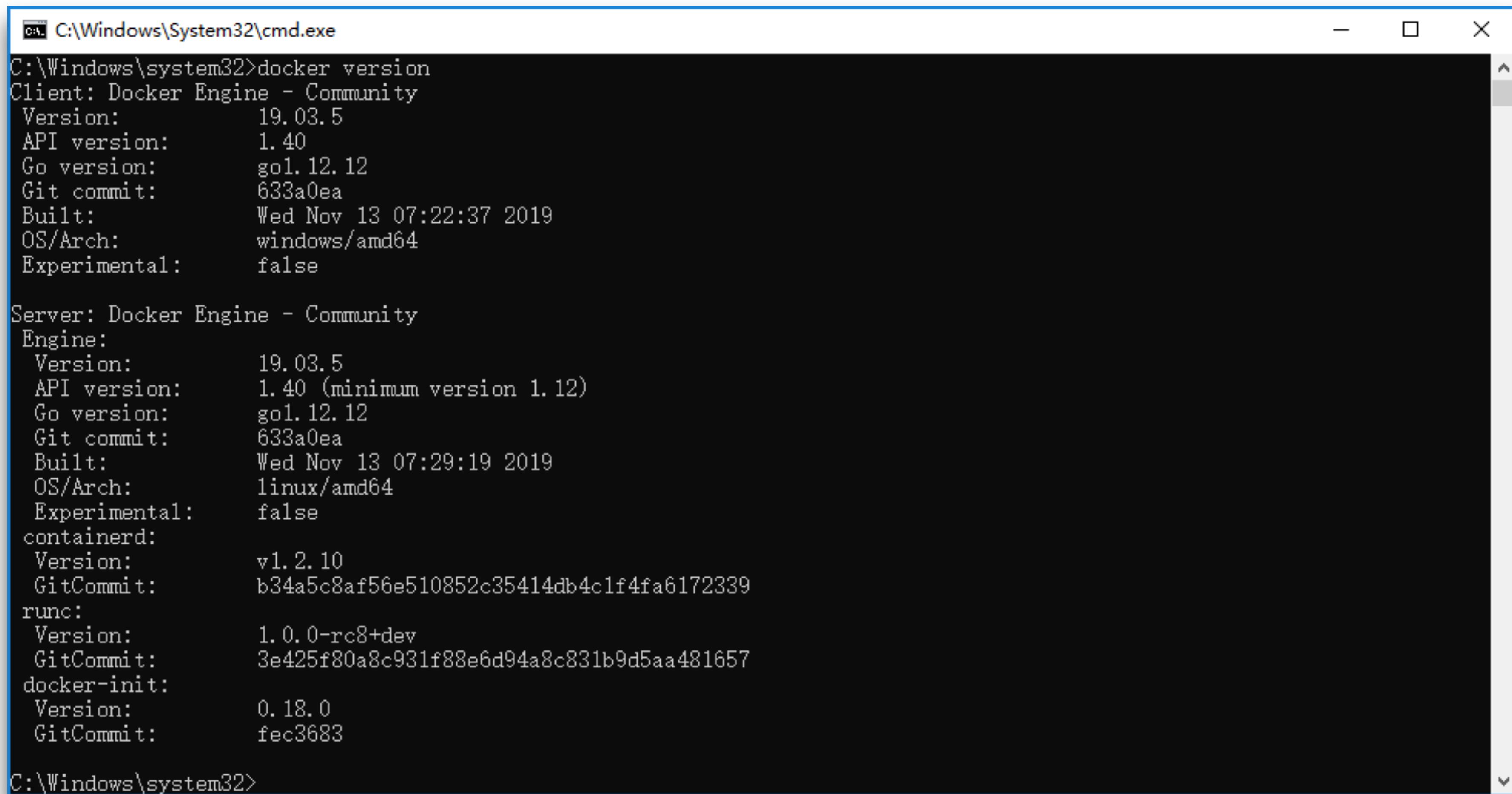


镜像加速器设置

The screenshot shows the Docker Settings interface. The left sidebar has a 'Settings' header and five categories: General, Resources, Docker Engine (which is selected and highlighted in blue), Command Line, and Kubernetes. The main content area is titled 'Docker Engine v19.03.5'. It contains instructions to configure the Docker daemon by typing a json Docker daemon configuration file, with a note that this can prevent Docker from starting. A code block shows the configuration snippet:

```
{  
  "registry-mirrors": ["https://docker.mirrors.ustc.edu.cn"],  
  "insecure-registries": [],  
  "debug": true,  
  "experimental": false  
}
```

校验安装 ~ docker version

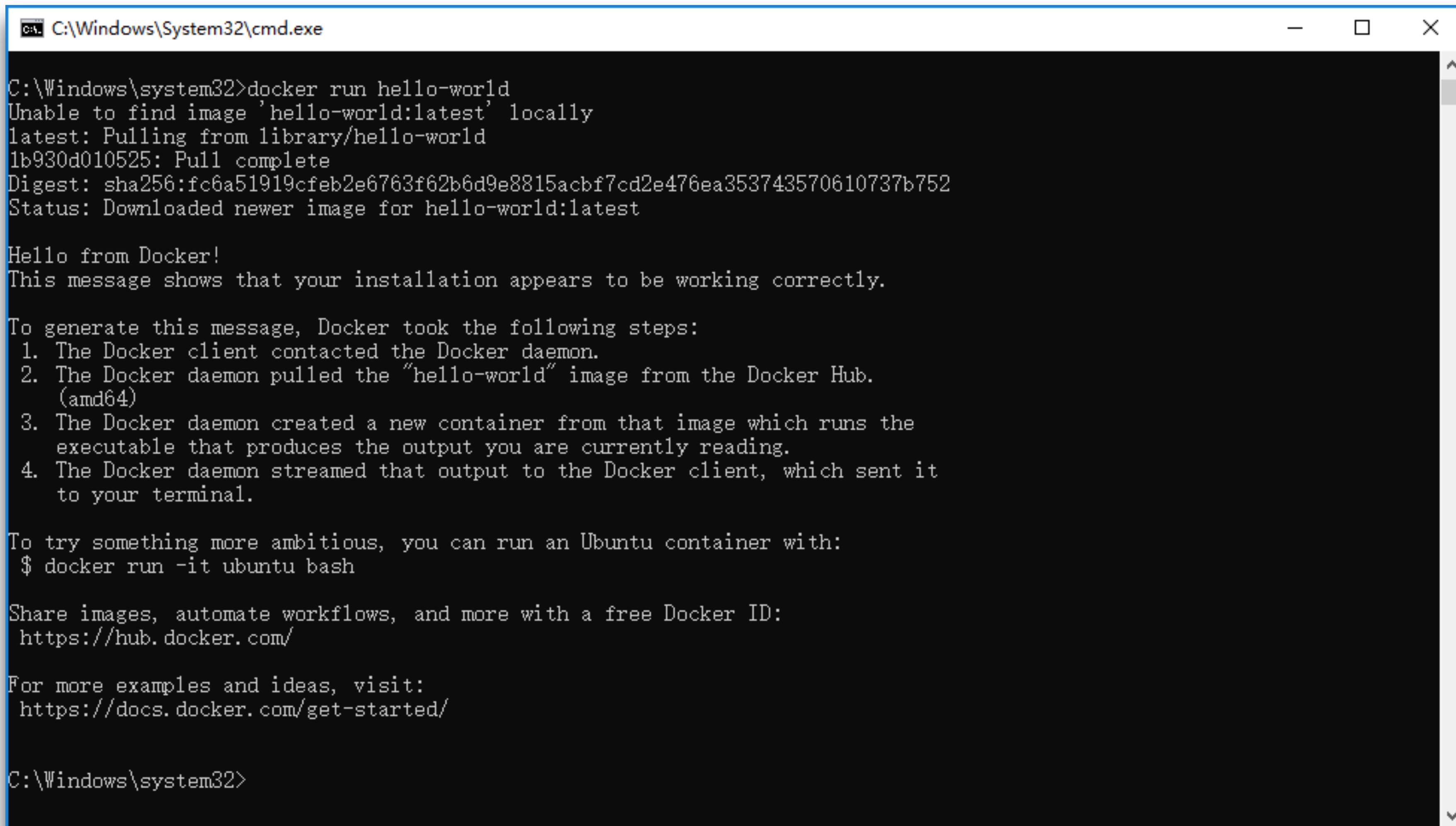


```
C:\Windows\System32\cmd.exe
C:\Windows\system32>docker version
Client: Docker Engine - Community
  Version:           19.03.5
  API version:      1.40
  Go version:       go1.12.12
  Git commit:       633a0ea
  Built:            Wed Nov 13 07:22:37 2019
  OS/Arch:          windows/amd64
  Experimental:    false

Server: Docker Engine - Community
  Engine:
    Version:          19.03.5
    API version:     1.40 (minimum version 1.12)
    Go version:      go1.12.12
    Git commit:      633a0ea
    Built:           Wed Nov 13 07:29:19 2019
    OS/Arch:         linux/amd64
    Experimental:   false
  containerd:
    Version:          v1.2.10
    GitCommit:        b34a5c8af56e510852c35414db4clf4fa6172339
  runc:
    Version:          1.0.0-rc8+dev
    GitCommit:        3e425f80a8c931f88e6d94a8c831b9d5aa481657
  docker-init:
    Version:          0.18.0
    GitCommit:        fec3683

C:\Windows\system32>
```

校验安装 ~ docker run hello-world



```
C:\Windows\system32>docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
1b930d010525: Pull complete
Digest: sha256:fc6a51919cfecb2e6763f62b6d9e8815acbf7cd2e476ea353743570610737b752
Status: Downloaded newer image for hello-world:latest

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/

C:\Windows\system32>
```

预加载k8s相关镜像

AliyunContainerService / k8s-for-docker-desktop

80 commits 13 branches 0 packages 0 releases 10 contributors

Branch: master New pull request

denverdino Update README.md

images Refine the README for dashboard configuration

sample Add ingress

README.md Update README.md

README_en.md Update for v1.14.8

getLatestIstio.ps1 'get istio version automatically'

images.properties Update to K8s v1.15.5

kubernetes-dashboard.yaml Update for dashboard 1.10.1

load_images.ps1 Update windows support script

load_images.sh fix shebang

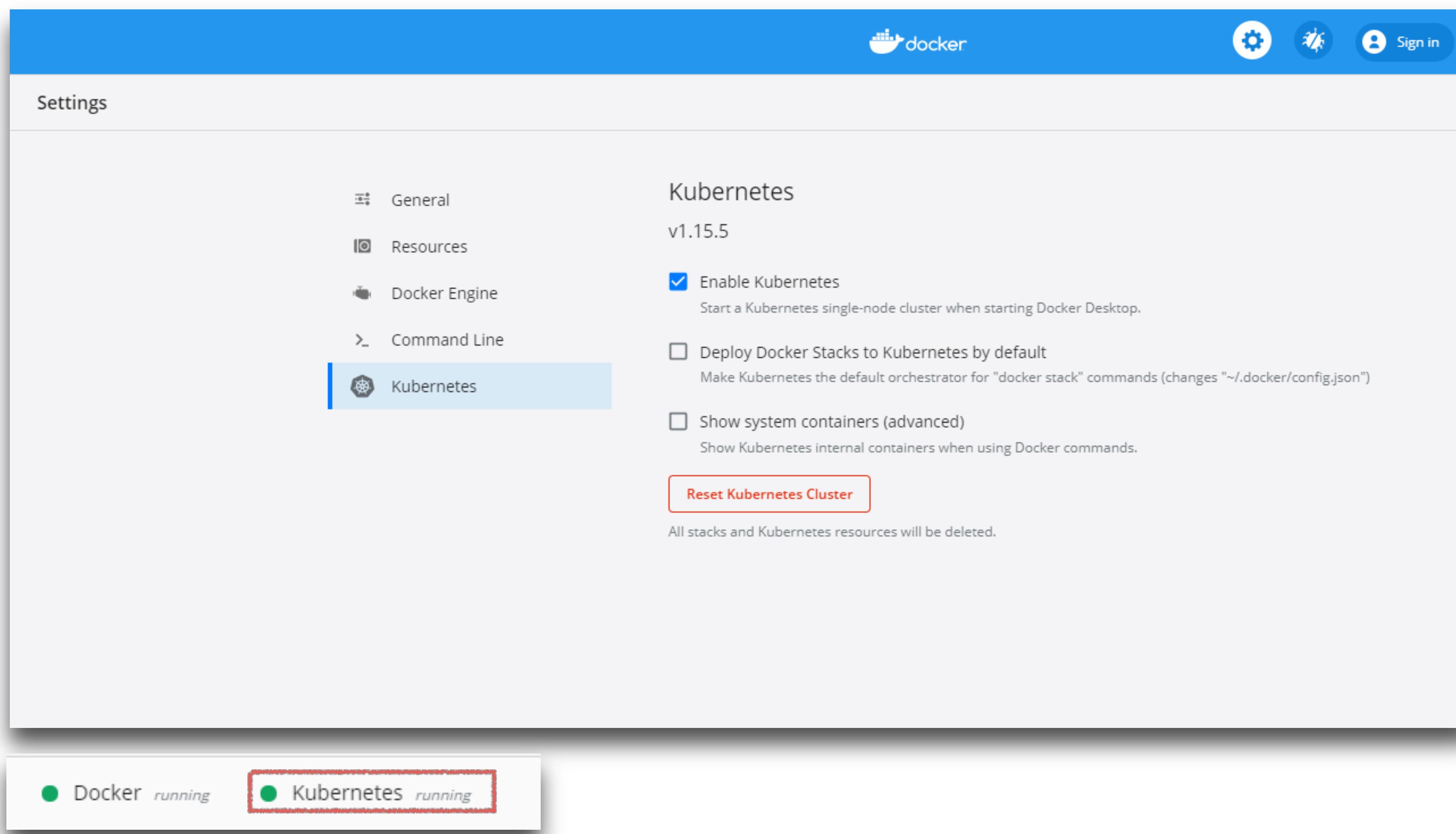
README.md

```
选择Windows PowerShell
PS D:\temp\k8s-for-docker-desktop> .\load_images.ps1
k8s.gcr.io/pause:3.1=registry.cn-hangzhou.aliyuncs.com/google_containers/pause:3.1
3.1: Pulling from google_containers/pause
cf9202429979: Pull complete
Digest: sha256:759c3f0f6493093a9043cc813092290af69029699ade0e3dbe024e968fcb7cca
Status: Downloaded newer image for registry.cn-hangzhou.aliyuncs.com/google_containers/pause:3.1
registry.cn-hangzhou.aliyuncs.com/google_containers/pause:3.1
Untagged: registry.cn-hangzhou.aliyuncs.com/google_containers/pause:3.1
Untagged: registry.cn-hangzhou.aliyuncs.com/google_containers/pause@sha256:759c3f0f6493093a9043cc813092290af69029699ade
0e3dbe024e968fcb7cca
k8s.gcr.io/kube-controller-manager:v1.15.5=registry.cn-hangzhou.aliyuncs.com/google_containers/kube-controller-manager:
v1.15.5
v1.15.5: Pulling from google_containers/kube-controller-manager
39fafc05754f: Downloading [=====] 7.864MB/17.74MB
dfc48811dd58: Downloading [=====] 19.1MB/30.11MB
```

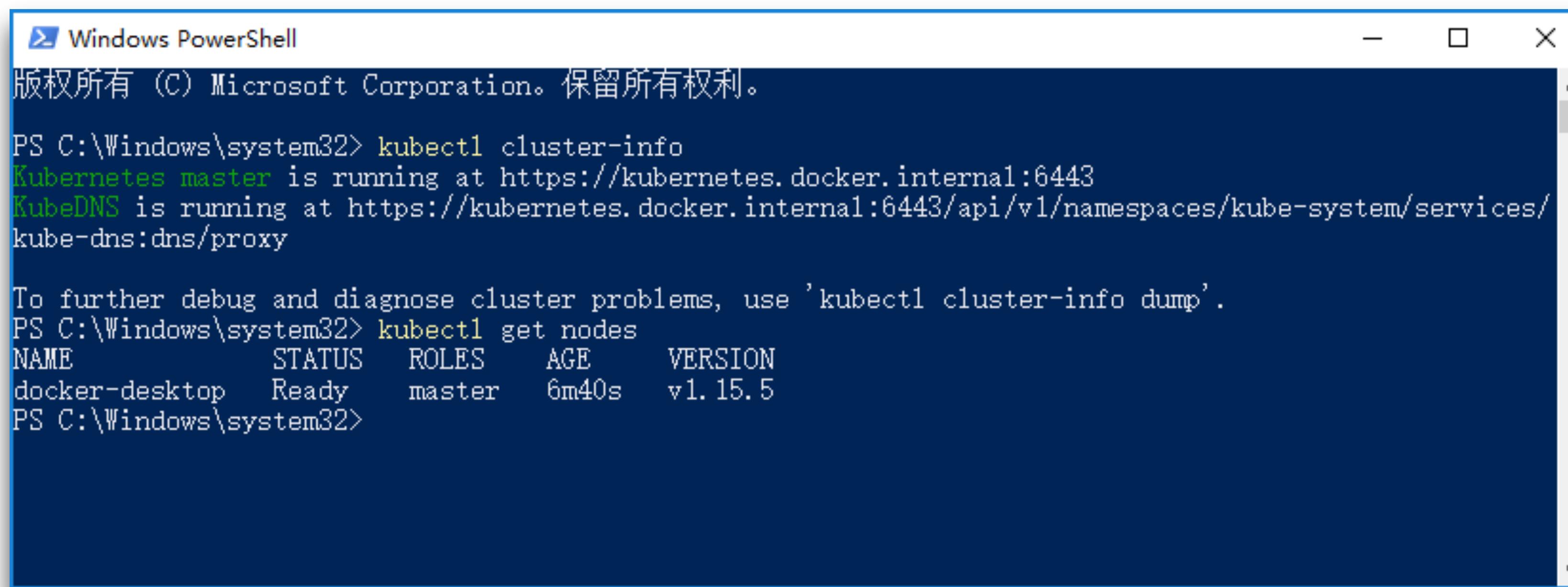
```
Windows PowerShell
Status: Downloaded newer image for registry.cn-hangzhou.aliyuncs.com/google_containers/nginx-ingress-controller:0.26.1
registry.cn-hangzhou.aliyuncs.com/google_containers/nginx-ingress-controller:0.26.1
Untagged: registry.cn-hangzhou.aliyuncs.com/google_containers/nginx-ingress-controller:0.26.1
Untagged: registry.cn-hangzhou.aliyuncs.com/google_containers/nginx-ingress-controller@sha256:5da1b2e84ecbdb27facbea84bc6ddc9d50145d8249
63230735b47828891cba7b
PS D:\temp\k8s-for-docker-desktop> docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
k8s.gcr.io/kube-controller-manager v1.15.5 1399a72fa1a9 4 months ago 159MB
k8s.gcr.io/kube-proxy v1.15.5 cbd7f21fec99 4 months ago 82.4MB
k8s.gcr.io/kube-apiserver v1.15.5 e534b1952a0d 4 months ago 207MB
k8s.gcr.io/kube-scheduler v1.15.5 fab2dded59dd 4 months ago 81.1MB
quay.io/kubernetes-ingress-controller/nginx-ingress-controller 0.26.1 29024c9c6e70 5 months ago 483MB
k8s.gcr.io/coredns 1.3.1 eb516548c180 13 months ago 40.3MB
hello-world latest fce289e99eb9 14 months ago 1.84kB
k8s.gcr.io/kubernetes-dashboard-amd64 v1.10.1 f9aed6605b81 14 months ago 122MB
k8s.gcr.io/etcd 3.3.10 2c4adeb21b4f 15 months ago 258MB
k8s.gcr.io/pause 3.1 da86e6ba6ca1 2 years ago 742kB
PS D:\temp\k8s-for-docker-desktop>
```

<https://github.com/AliyunContainerService/k8s-for-docker-desktop>

启用K8s设置



校验K8s安装



A screenshot of a Windows PowerShell window titled "Windows PowerShell". The window shows the output of several commands related to Kubernetes:

```
版权所有 (C) Microsoft Corporation。保留所有权利。
PS C:\Windows\system32> kubectl cluster-info
Kubernetes master is running at https://kubernetes.docker.internal:6443
KubeDNS is running at https://kubernetes.docker.internal:6443/api/v1/namespaces/kube-system/services/
kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
PS C:\Windows\system32> kubectl get nodes
NAME           STATUS    ROLES   AGE     VERSION
docker-desktop  Ready     master   6m40s   v1.15.5
PS C:\Windows\system32>
```

安装k8s dashboard并启动proxy

```
PS C:\Windows\system32> kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/v1.10.1/src/deploy/recommended/kubernetes-dashboard.yaml
secret/kubernetes-dashboard-certs created
serviceaccount/kubernetes-dashboard created
role.rbac.authorization.k8s.io/kubernetes-dashboard-minimal created
rolebinding.rbac.authorization.k8s.io/kubernetes-dashboard-minimal created
deployment.apps/kubernetes-dashboard created
service/kubernetes-dashboard created
PS C:\Windows\system32> kubectl get ns
NAME      STATUS   AGE
default   Active   8m26s
docker    Active   7m7s
kube-node-lease Active  8m27s
kube-public Active  8m27s
kube-system Active  8m27s
PS C:\Windows\system32> kubectl get all -n kube-system
NAME                           READY   STATUS    RESTARTS   AGE
pod/coredns-5c98db65d4-1w29m   1/1    Running   0          8m24s
pod/coredns-5c98db65d4-vwjw2   1/1    Running   0          8m24s
pod/etcfd-docker-desktop       1/1    Running   0          7m8s
pod/kube-apiserver-docker-desktop 1/1    Running   0          7m16s
pod/kube-controller-manager-docker-desktop 1/1    Running   0          7m33s
pod/kube-proxy-6pd8p           1/1    Running   0          8m25s
pod/kube-scheduler-docker-desktop 1/1    Running   0          7m13s
pod/kubernetes-dashboard-7d75c474bb-pscjh 1/1    Running   0          20s

NAME            TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)          AGE
service/kube-dns ClusterIP   10.96.0.10   <none>        53/UDP, 53/TCP, 9153/TCP 8m33s
service/kubernetes-dashboard ClusterIP  10.108.11.124 <none>        443/TCP          20s

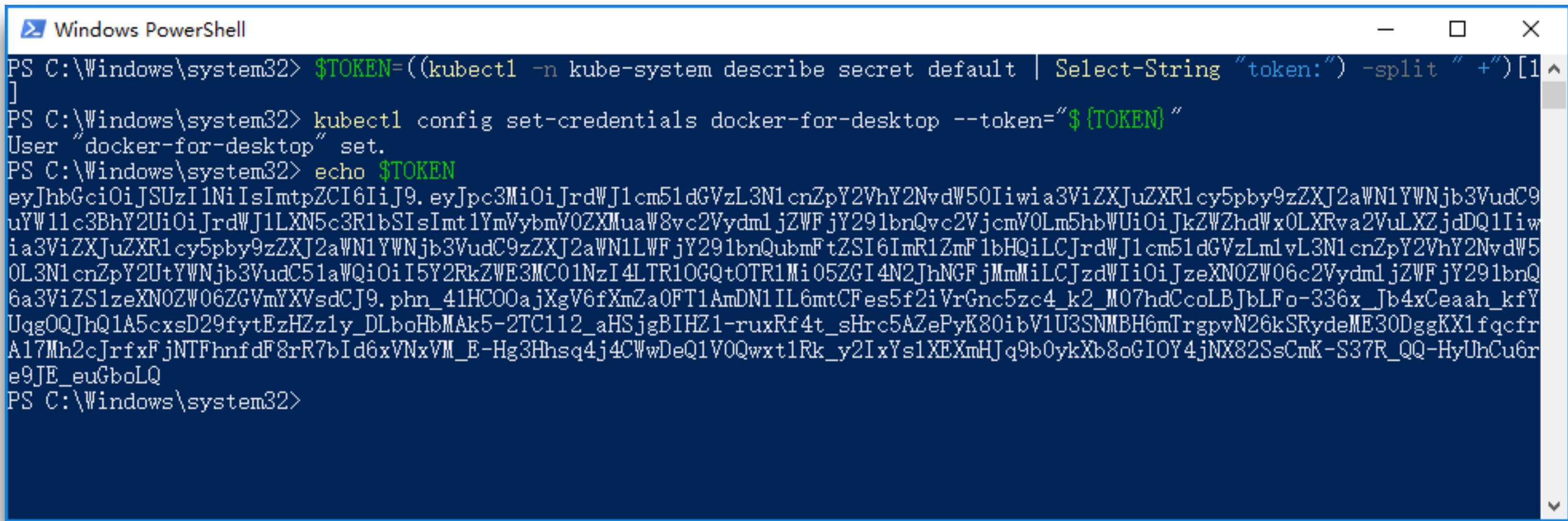
NAME           DESIRED   CURRENT   READY   UP-TO-DATE   AVAILABLE   NODE SELECTOR   AGE
daemonset.apps/kube-proxy  1         1         1        1           1           beta.kubernetes.io/os=linux 8m33s

NAME                           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/coredns        2/2     2           2           8m33s
deployment.apps/kubernetes-dashboard 1/1     1           1           20s

NAME           DESIRED   CURRENT   READY   AGE
replicaset.apps/coredns-5c98db65d4  2         2         2           8m24s
replicaset.apps/kubernetes-dashboard-7d75c474bb 1         1         1           20s

PS C:\Windows\system32> kubectl proxy
Starting to serve on 127.0.0.1:8001
```

获取dashboard令牌

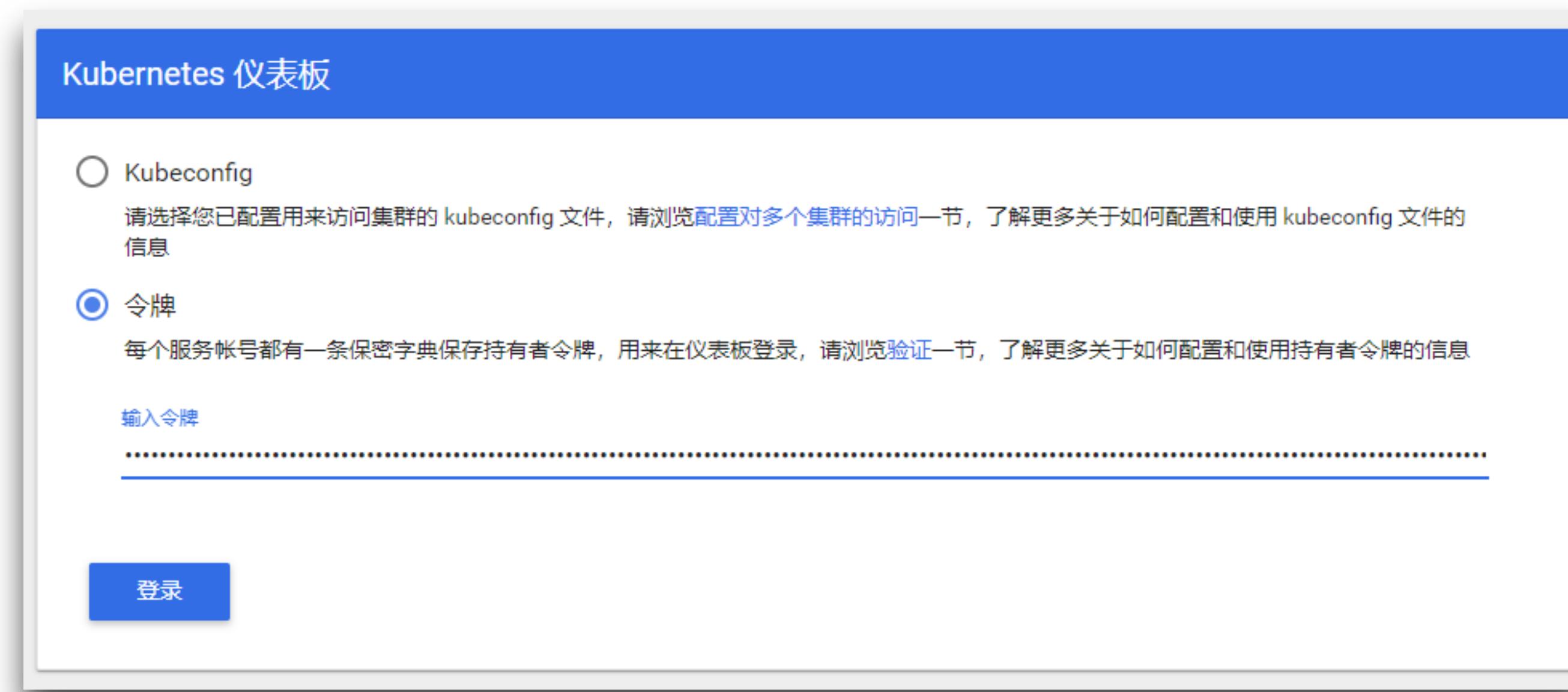


A screenshot of a Windows PowerShell window titled "Windows PowerShell". The window shows the following command sequence:

```
PS C:\Windows\system32> $TOKEN=((kubectl -n kube-system describe secret default | Select-String "token:") -split " +")[1]
PS C:\Windows\system32> kubectl config set-credentials docker-for-desktop --token="$TOKEN"
User "docker-for-desktop" set.
PS C:\Windows\system32> echo $TOKEN
eyJhbGciOiJSUzI1NiIsImtpZCI6IiJ9.eyJpc3MiOiJrdWJ1cm51dGVzL3N1cnZpY2VhY2NvdW50Iiwia3ViZXJuZXR1cy5pbv9zZXJ2aWN1YWNjb3VudC9uYW11c3BhY2UiOjJrdWJ1LXN5c3R1bSIisImt1YmVybmV0ZXMuaw8vc2VydmljZWFljY291bnQvc2VjcmV0Lm5hbWUiOjJkZWZhdWxOLXRva2VuLXZjdDQ1Iiwi3ViZXJuZXR1cy5pbv9zZXJ2aWN1YWNjb3VudC9zZXJ2aWN1LWFjY291bnQubmFtZSI6ImR1ZmF1bHQiLCJrdWJ1cm51dGVzLm1vL3N1cnZpY2VhY2NvdW50L3N1cnZpY2UtYWNjb3VudC51aWQiOjI5Y2RkZWE3MC01NzI4LTr1OGQtOTR1Mi05ZGI4N2JhNGFjMmMilCJzdWIiOjzeXN0ZW06c2VydmljZWFljY291bnQ6a3ViZS1zeXN0ZW06ZGVmYXVsDCJ9.phn_41HC00ajXgV6fXmZa0FT1AmDN1IL6mtCFes5f2iVrGnc5zc4_k2_M07hdCcoLBJbLFo-336x_Jb4xCeaah_kfYUqg0QJhQ1A5cxsD29fytEzHZz1y_DLboHbMAk5-2TC112_aHSjgBIHZ1-ruxRf4t_sHrc5AZePyK80ibV1U3SNMBH6mTrgpvN26kSRydeME30DggKX1fqcfrA17Mh2cJrfxFjNTFhndF8rR7bId6xVNxVM_E-Hg3Hhsq4j4CWwDeQ1V0Qwxt1Rk_y2IxYs1XEXmHJq9b0ykXb8oGIOY4jNX82SsCmK-S37R_QQ-HyUhCu6re9JE_euGboLQ
PS C:\Windows\system32>
```

```
$TOKEN=((kubectl -n kube-system describe secret default | Select-String "token:") -split " +")[1]
kubectl config set-credentials docker-for-desktop --token="$TOKEN"
echo $TOKEN
```

登录dashboard



<http://localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/#!/overview?namespace=default>

访问dashboard

docker-desktop - Kubernetes | +

localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy#!/node/docker-desktop?namespace=default

kubernetes

Search

+ 创建 | 🔍

集群 > 节点 > docker-desktop

集群

命名空间

节点

持久化存储卷

角色

存储类

命名空间

default

概况

工作负载

定时任务

守护进程集

部署

任务

容器组

副本集

副本控制器

有状态副本集

服务发现与负载均衡

访问权

服务

配置与存储

更多

详情

名称: docker-desktop

标签:

beta.kubernetes.io/arch: amd64
beta.kubernetes.io/os: linux
kubernetes.io/arch: amd64
kubernetes.io/hostname: docker-desktop

注释:

kubeadm.alpha.kubernetes.io/cri-socket: /var/run/dockershim.sock
node.alpha.kubernetes.io/ttl: 0
volumes.kubernetes.io/controller-managed-attach-detach: true

系统信息

机器 ID: eb30b84e-08f5-4473-8499-ecfde6b1d0b1

系统 UUID: b5694e03-877a-4888-abebe5d673c26c32

启动 ID: f36a50ea-14f9-4e77-a4d8-33dc8edfd207

内核版本: 4.19.76-linuxkit

操作系统镜像: Docker Desktop

容器运行时版本: docker://19.3.5

Kubelet 版本: v1.15.5

Kube-Proxy 版本: v1.15.5

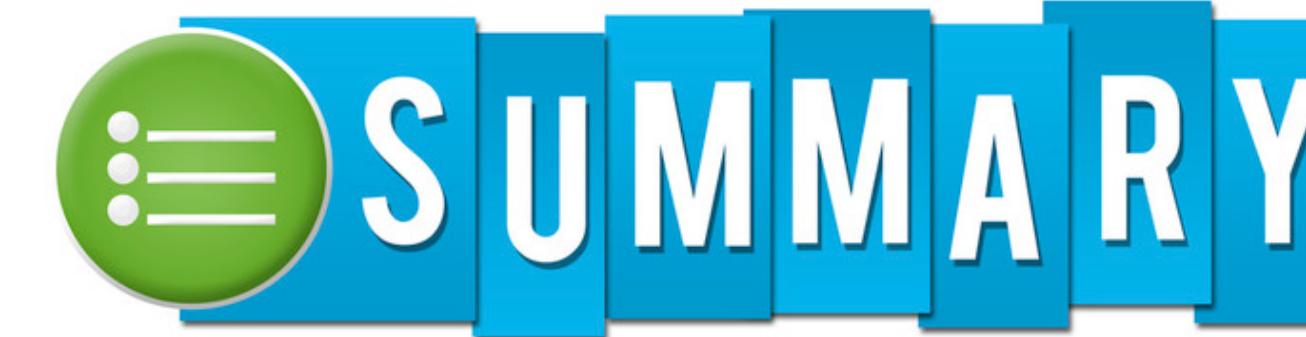
操作系统: linux

架构: amd64

已分配资源

Docker Desktop....exe 全部显示

本课小结



- 演示本地安装Docker Desktop for Windows + 启用K8s
 - 先决条件：Win10版本，虚拟化启用，否则只能考虑Minikube + VirtualBox
 - 设置容器资源(CPU/Memory)
 - 设置镜像加速器(考虑申请阿里云镜像加速器)
 - 阿里云k8s镜像预加载脚本(国内网络)
 - 安装参考：<https://github.com/AliyunContainerService/k8s-for-docker-desktop>
 - 安装步骤如更新请参考官方安装文档：<https://docs.docker.com/docker-for-windows/install/>