

Windows 上的 Docker 引擎

2016/08/23 作者 

本文内容

[安装 Docker](#)

[使用配置文件配置 Docker](#)

[在 Docker 服务上配置 Docker](#)

[通用配置](#)

[代理配置](#)

[卸载 Docker](#)

Docker 引擎和客户端包含在 Windows 并不需要进行安装和单独配置。此外，Docker 引擎可以接受多种自定义配置。例如，可以配置守护程序接受传入请求的方式、默认网络选项及调试/日志设置。在 Windows 上，这些配置可以在配置文件中指定，或者通过使用 Windows 服务控制管理器指定。本文档详细介绍了如何安装和配置 Docker 引擎，并且还提供了一些常用配置的示例。

安装 Docker

若要使用 Window 容器，则需要安装 Docker。Docker 由 Docker 引擎 (dockerd.exe) 和 Docker 客户端 (docker.exe) 组成。快速入门指南中提供了安装所有内容的最简方法。它们可以帮助你获得的所有内容设置和运行你的第一个容器。

- [Windows Server 2016 上的 Windows 容器](#)
- [Windows 10 上的 Windows 容器](#)

若要进行脚本化安装，请参阅[使用脚本安装 Docker EE](#)。

可以使用 Docker 容器映像需要先进行安装。有关详细信息，请参阅[映像使用快速入门指南](#)。

使用配置文件配置 Docker

在 Windows 上配置 Docker 引擎的首选方法是使用配置文件。可在“C:\ProgramData\Docker\config\daemon.json”中找到配置文件。如果此文件还不存在，可以创建此文件。

注意：并非所有可用的 Docker 配置选项在 Windows 上的 Docker 中都适用。以下示例列出了可用的选项。有关 Docker 引擎配置的完整文档，请参阅 [Docker 守护程序配置文件](#)。

 复制

```
{
  "authorization-plugins": [],
  "dns": [],
  "dns-opts": [],
```

```

"dns-search": [],
"exec-opts": [],
"storage-driver": "",
"storage-opts": [],
"labels": [],
"log-driver": "",
"mtu": 0,
"pidfile": "",
"data-root": "",
"cluster-store": "",
"cluster-advertise": "",
"debug": true,
"hosts": [],
"log-level": "",
"tlsverify": true,
"tlscacert": "",
"tlscert": "",
"tlskey": "",
"group": "",
"default-ulimits": {},
"bridge": "",
"fixed-cidr": "",
"raw-logs": false,
"registry-mirrors": [],
"insecure-registries": [],
"disable-legacy-registry": false
}

```

只需将想要进行的配置更改添加到配置文件即可。例如，此示例中将 Docker 引擎配置为接受端口 2375 传入的连接。其他所有配置选项将使用默认值。

复制

```

{
  "hosts": ["tcp://0.0.0.0:2375"]
}

```

同样，此示例将配置 Docker 守护程序以将图像和容器保存在备用路径。如果未指定，默认路径为 c:\programdata\docker。

复制

```

{
  "data-root": "d:\\docker"
}

```

同样，此示例将 Docker 守护程序配置为仅接受通过端口 2376 的安全连接。

复制

```

{
  "hosts": ["tcp://0.0.0.0:2376", "npipe://"],

```

```
"tlsverify": true,
"tlscacert": "C:\\ProgramData\\docker\\certs.d\\ca.pem",
"tlscert": "C:\\ProgramData\\docker\\certs.d\\server-cert.pem",
"tlskey": "C:\\ProgramData\\docker\\certs.d\\server-key.pem",
}
```

在 Docker 服务上配置 Docker

还可以通过使用 `sc config` 修改 Docker 服务来配置 Docker 引擎。使用此方法时将直接在 Docker 服务上设置 Docker 引擎的标记。在命令提示符 (cmd.exe 而非 PowerShell) 中运行以下命令：

```
sc config docker binpath= "\"C:\\Program Files\\docker\\dockerd.exe\" --run-service -H
tcp://0.0.0.0:2375"
```

注意：如果 daemon.json 文件已经包含 `"hosts": ["tcp://0.0.0.0:2375"]` 条目，则无需运行此命令。

通用配置

以下配置文件示例演示了通用的 Docker 配置。这些配置可以并入单个配置文件。

创建默认网络

若要将 Docker 引擎配置为不创建默认 NAT 网络，请运行以下操作。有关详细信息，请参阅[管理 Docker 网络](#)。

```
{
  "bridge" : "none"
}
```

设置 Docker 安全组

登录到 Docker 主机并在本地运行 Docker 命令后，这些命令将通过命名管道运行。默认情况下，只有管理员组的成员才可以通过此命名管道访问 Docker 引擎。若要指定具有此访问权限的安全组，请使用 `group` 标记。

```
{
  "group" : "docker"
}
```

代理配置

若要设置 `docker search` 和 `docker pull` 的代理信息，请使用 `HTTP_PROXY` 或 `HTTPS_PROXY` 名称以及代理信息的一个值创建 Windows 环境变量。可使用类似于以下的命令通过 PowerShell 完成此操作：

PowerShell	复制
<pre>[Environment]::SetEnvironmentVariable("HTTP_PROXY", "http://username:password@proxy:port/", [EnvironmentVariableTarget]::Machine)</pre>	

设置变量后，重启 Docker 服务。

PowerShell	复制
<pre>Restart-Service docker</pre>	

有关详细信息，请参阅 [Docker.com 上的 Windows 配置文件](#)。

卸载 Docker

使用本部分中的步骤卸载 Docker，并全面清理 Windows 10 或 Windows Server 2016 系统中的 Docker 系统组件。

注意：以下步骤中的所有命令都必须从**提升的** PowerShell 会话中运行。

步骤 1: 准备你的系统以删除 Docker

最好在删除 Docker 之前确保系统上未运行容器。下面是用于执行此操作的一些有效命令：

	复制
<pre># Leave swarm mode (this will automatically stop and remove services and overlay networks) docker swarm leave --force # Stop all running containers docker ps --quiet ForEach-Object {docker stop \$_}</pre>	

最好在删除 Docker 之前也从系统中删除所有容器、容器映像、网络 and 卷：

	复制
<pre>docker system prune --volumes --all</pre>	

步骤 2: 卸载 Docker

在 Windows 10 上卸载 Docker 的步骤: 10:

- 在 Windows 10 计算机上转到“设置”>“应用”
- 在“应用和功能”下面，查找“适用于 Windows 的 Docker”
- 单击“适用于 Windows 的 Docker”>“卸载”

在 Windows Server 2016 上卸载 Docker 的步骤: 16:

从提升的 PowerShell 会话中，使用 `Uninstall-Package` 和 `Uninstall-Module` cmdlet 从系统中删除 Docker 模块及其相应的程序包管理提供程序。

提示：你可以查找曾用于安装 Docker 的程序包提供程序 `PS C:\> Get-PackageProvider -Name *Docker*`

例如:

```
Uninstall-Package -Name docker -ProviderName DockerMsftProvider
Uninstall-Module -Name DockerMsftProvider
```

步骤 3: 清理 Docker 数据和系统组件

删除 Docker 的默认网络，以便在卸载 Docker 的同时在你的系统上清除这些网络的配置:

```
Get-HNSNetwork | Remove-HNSNetwork
```

从系统中删除 Docker 的程序数据:

```
Remove-Item "C:\ProgramData\Docker" -Recurse
```

你可能还需要删除 Windows 上与 Docker/容器关联的 Windows 可选功能。

至少，这包括“容器”功能，安装 Docker 时会在任何 Windows 10 或 Windows Server 2016 上自动启用该功能。这还可能包括“Hyper-V”功能，安装 Docker 时可在 Windows 10 上自动启用该功能，但必须在 Windows Server 2016 上显式启用该功能。

关于禁用 HYPER-V 的重要说明: [Hyper-V 功能](#)是一种常规虚拟化功能，该功能所启用的远远不止是容器！禁用 Hyper-V 功能之前，请确保系统上没有其他虚拟化组件需要该功能。

在 Windows 10 上删除 Windows 功能的步骤: 10:

- 在 Windows 10 计算机上，转到“控制面板”>“程序”>“程序和功能”>“打开或关闭 Windows 功能”
- 查找你想要禁用的功能的名称，在本例中为“容器”和（可选）“Hyper-V”
- **取消选中**你想要禁用的功能名称旁边的框
- 单击“确定”

在 Windows Server 2016 上删除 Windows 功能的步骤：16:

从提升的 PowerShell 会话中，使用以下命令禁用系统中的“容器”和（可选）“Hyper-V”功能：

 复制
<pre>Remove-WindowsFeature Containers Remove-WindowsFeature Hyper-V</pre>

步骤 4：重新启动你的系统

若要完成这些卸载/清理步骤，请从提升的 PowerShell 会话中运行：

 复制
<pre>Restart-Computer -Force</pre>