



# KubeSphere 企业版

## 安装和卸载

文档版本：01

发布日期：2023-08-21

## 【版权声明】

版权所有 © 北京青云科技股份有限公司 2023。保留所有权利。

未经本公司书面许可，任何单位和个人不得摘抄、复制或以其他方式传播本文档的部分或全部内容。

## 【商标声明】

 和其他青云商标均由北京青云科技股份有限公司拥有。

本文档提及的其他商标或注册商标，由各自的所有人拥有。

## 【产品或服务声明】

本文档描述的部分产品、服务和特性可能不在您的购买或使用范围之内，您购买的产品、服务或特性应受青云公司商业合同和条款的约束。

本文档内容会不定期进行更新。本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

北京青云科技股份有限公司

网址：<https://www.qingcloud.com>

# 前言

本节介绍如何安装和卸载 KubeSphere 企业版的核心组件 KubeSphere Core。

KubeSphere Core 安装完成后，即可使用 KubeSphere 企业版。

**备注**

暂不支持从 KubeSphere 企业版 3.x 版本升级到 4.x，将在后续版本中支持 3.x 版本的升级。

# 目录

---

1. 安装 KubeSphere 企业版 .....	1
2. 部署 KubeSphere 企业版 .....	4
3. 卸载 KubeSphere 企业版 .....	10

# 1. 安装 KubeSphere 企业版

本节介绍如何安装 KubeSphere 企业版 v4.0。

安装过程中将用到开源工具 KubeKey。有关安装的更多信息，请访问 [GitHub KubeKey 仓库](#)。

## 前提条件

您需要准备一台 Linux 主机，并确保主机软硬件满足以下条件：

类别	配置要求
硬件	最低配置：2 核 CPU，2 GB 内存，40 GB 磁盘空间。
操作系统	<ul style="list-style-type: none"><li>• Ubuntu 16.04, 18.04, 20.04, 22.04</li><li>• Debian Bullseye, Buster, Stretch</li><li>• CentOS/RHEL 7</li><li>• AlmaLinux 9.0</li><li>• SUSE Linux Enterprise Server 15</li></ul>
Linux 内核	4.15 及更高版本。可以通过命令 <code>uname -srnm</code> 查看 Linux 内核版本。
网络	<ul style="list-style-type: none"><li>• 请确保 <code>/etc/resolv.conf</code> 中的 DNS 地址可用。</li><li>• 建议禁用防火墙以避免组件通信受阻。</li></ul>
其他	节点上可以使用 <code>sudo</code> 、 <code>curl</code> 和 <code>openssl</code> 命令。

## 安装 Kubernetes

安装 KubeSphere 企业版之前，你需要一个可用的 Kubernetes 集群。执行以下命令快速创建 Kubernetes 集群。

KubeSphere 企业版支持的 Kubernetes 版本有：v1.21.x、v1.22.x、v1.23.x、v1.24.x、v1.25.x、v1.26.x 和 v1.27.x。最低支持版本为 v1.21。

备注

KubeSphere 企业版中各个扩展组件支持的 Kubernetes 版本和 KubeSphere 企业版支持的 Kubernetes 版本可能不同。安装扩展组件时，请以扩展市场的组件详情页中的 **Kubernetes 版本** 为准。

1. 如果您访问 GitHub/Googleapis 受限，请登录 Linux 主机，执行以下命令设置下载区域。

```
export KKZONE=cn
```

2. 执行以下命令下载安装工具 KubeKey。

```
curl -sfL https://get-kk.kubesphere.io | sh -
```

**备注**

KubeKey 是 Kubernetes 和 KubeSphere 的安装工具。有关 KubeKey 的更多信息，请访问 [KubeKey GitHub 仓库](#)。

3. 执行以下命令安装依赖项 socat 和 conntrack。

```
apt install socat conntrack -y
```

4. 执行以下命令快速创建一个 Kubernetes 集群。

```
./kk create cluster --skip-pull-images --with-local-storage --with-kubernetes v1.25.4  
--container-manager containerd -y
```

## 安装 KubeSphere 企业版

您需要提前安装 Helm。具体请参阅[安装 Helm](#)。

1. 如果您已经拥有可用的 Kubernetes 集群，执行以下命令通过 [helm](#) 安装 KubeSphere Core。

```
helm upgrade --install -n kubesphere-system --create-namespace ks-core  
https://charts.kubesphere.io/main/ks-core-0.4.0.tgz
```

如果显示如下信息，则表明安装成功：

**NOTES:**

Please wait for several seconds for KubeSphere deployment to complete.

1. Make sure KubeSphere components are running:

```
kubectl get pods -n kubesphere-system
```

2. Then you should be able to visit the console NodePort:

Console: <http://172.31.1.1:30880>

3. To login to your KubeSphere console:

Account: admin

Password: "P@88w0rd"

NOTE: Please change the default password after login.

For more details, please visit <https://kubesphere.io>.

2. 执行以下命令检查 Pod 状态。

```
kubectl get pod -n kubesphere-system
```

3. 当 Pod 状态都为 **Running** 时，从安装成功信息中的 Console、Account 和 Password 参数分别获取 KubeSphere 企业版 Web 控制台的 IP 地址、管理员用户名和管理员密码，并使用网页浏览器登录 KubeSphere 企业版 Web 控制台。

备注

取决于您的网络环境，您可能需要配置流量转发规则并在防火墙中放行 30880 端口。

## 2. 部署 KubeSphere 企业版

本节介绍如何使用 Helm CLI 在 Kubernetes 集群上部署 KubeSphere 企业版 v4.0。

### 前提条件

- 可用的 Kubernetes 集群。KubeSphere 可以安装在任何 Kubernetes 集群上。有关如何创建 Kubernetes 集群，请参阅[安装 Kubernetes](#)。
- 已安装 Ingress Controller。

#### 备注

KubeSphere 企业版 Web 控制台（**Console UI**）通过 **NodePort** 或者 **Ingress** 来支持外部访问。若要通过 Ingress 暴露 Console 服务，必须在 Kubernetes 集群安装 Ingress Controller。

对于默认不包含 Ingress Controller 的发行版（例如 EKS、GKE 或 AKS 等托管 Kubernetes 集群），必须先部署 Ingress Controller。

请参考以下教程安装 Ingress Controller：

<https://kubernetes.github.io/ingress-nginx/deploy/>

<https://doc.traefik.io/traefik/getting-started/install-traefik/#use-the-helm-chart>

- 已安装 CLI 工具。

#### 备注

配置 Kubernetes 集群需要使用以下 CLI 工具。请确保这些工具已安装，并在您的 `$PATH` 中可用。

- [kubect](#)l：Kubernetes 命令行工具。
- [helm](#)：Kubernetes 的包管理器。在不同系统上的安装说明，请参考 [Helm 安装文档](#)。

### 操作步骤

#### 1. 通过 `helm` 安装 KubeSphere Core

```
helm upgrade --install -n kubespere-system --create-namespace ks-core
https://charts.kubesphere.io/main/ks-core-0.4.0.tgz
```

#### 2. 选择 SSL 配置

KubeSphere 安全配置分为**网关 SSL 配置**以及**内部服务 SSL 配置**两个部分。其中**网关 SSL 配置**默认支持以下三种模式来启用 SSL/TLS，以保证访问的安全性。，以确保访问的安全性。

##### a. 网关 SSL 配置



配置	Helm Chart 选项	是否需要 cert-manager
KubeSphere 生成的 TLS 证书	ingress.tls.source=generation	否
Let' s Encrypt	ingress.tls.source=letsEncrypt	是
导入已有的证书	ingress.tls.source=importation	否

- KubeSphere 生成的 TLS 证书：支持 cert-manager 和 helm 两种方式。
  - 如果 Kubernetes 集群中已安装 cert-manager，则首选使用 cert-manager 生成证书。  
KubeSphere 使用 cert-manager 签发并维护证书。KubeSphere 会生成自己的 CA 证书，并用该 CA 签署证书，然后由 cert-manager 管理该证书。
  - 如果未安装 cert-manager，则使用 helm 生成证书。在使用 helm 安装的过程中，KubeSphere 会根据设置的 `hostname` 生成 CA 和 TLS 证书。在此选项下，证书不支持自动过期轮转。
- Let' s Encrypt  
使用 Let' s Encrypt 选项必须使用 cert-manager。但是，在这种情况下，cert-manager 与 Let' s Encrypt 的特殊颁发者相结合，该颁发者执行获取 Let' s Encrypt 颁发证书所需的所有操作，包括请求和验证。此配置使用 HTTP 验证（HTTP-01），因此负载均衡器必须具有可以从互联网访问的公共 DNS 记录。
- 导入已有的证书  
使用已有的CA颁发的公有或私有证书。KubeSphere 将使用该证书来保护 WebSocket 和 HTTPS 流量。在这种情况下，您必须上传名称分别为 `tls.crt` 和 `tls.key` 的 PEM 格式的证书以及相关的密钥。  
如果您使用私有 CA，则还必须上传该 CA 证书。这是由于您的节点可能不信任此私有 CA。

#### b. 内部服务 SSL 配置

启用内部 SSL 配置之后，KubeSphere 中 Console UI 和 Apiserver 均会启用 HTTPS，内置支持 cert-manager 和 helm 生成证书。在 Kubernetes 集群已安装 cert-manager 的情况下优先使用 cert-manager 生成/管理证书，证书的 DNS 默认使用 Console UI 和 Apiserver 在 Kubernetes 集群内部的 Service DNS。

配置	Helm Chart 选项	是否需要 cert-manager
启用内部SSL	internalTLS=true	否

### 3. 安装 cert-manager

若使用自己的证书文件（`ingress.tls.source=importation`），您可以跳过此步骤。

仅在使用 KubeSphere 生成的证书（`ingress.tls.source=generation`）或 Let' s Encrypt 颁发的证书（`ingress.tls.source=letsEncrypt`）时，才需要安装 cert-manager。

```
# 添加 Jetstack Helm 仓库
helm repo add jetstack https://charts.jetstack.io

# 更新本地 Helm Chart 仓库缓存
helm repo update

# 安装 cert-manager Helm Chart
helm install cert-manager jetstack/cert-manager -n cert-manager --create-namespace --set prometheus.enabled=false
# 或
kubectl apply -f https://github.com/cert-manager/cert-manager/releases/download/<VERSION>/cert-manager.yaml
```

安装完 cert-manager 后，检查 cert-manager 命名空间中正在运行的 Pod 来验证它是否已正确部署：

```
kubectl get pods --namespace cert-manager
```

#### 4. 根据您选择的证书选项，通过 Helm 为 KubeSphere 开启 SSL 配置

##### a. 启用网关 SSL 配置

- KubeSphere 生成的证书

```
helm upgrade --install -n kubesphere-system --create-namespace ks-core
https://charts.kubesphere.io/main/ks-core-0.4.0.tgz \
--set ingress.enabled=true \
--set hostname=kubesphere.my.org
```

- Let's Encrypt

此选项使用 cert-manager 来自动请求和续订 Let's Encrypt 证书。Let's Encrypt 是免费的，而且是受信的 CA，因此可以提供有效的证书。

```
helm upgrade --install -n kubesphere-system --create-namespace ks-core
https://charts.kubesphere.io/main/ks-core-0.4.0.tgz \
--set hostname=kubesphere.my.org \
--set ingress.enabled=true \
--set ingress.tls.source=letsEncrypt \
--set letsEncrypt.email=me@example.org
```

- 导入外部证书

```
# 导入外部证书
kubectl create secret tls tls-ks-core-ingress --cert=tls.crt --key=tls.key -n
kubesphere-system

# 安装 KubeSphere
helm upgrade --install -n kubesphere-system --create-namespace ks-core
https://charts.kubesphere.io/main/ks-core-0.4.0.tgz \
--set ingress.enabled=true \
--set hostname=kubesphere.my.org \
--set ingress.tls.source=importation
```

##### b. 启用内部服务 SSL 配置

```
helm upgrade --install -n kubesphere-system --create-namespace ks-core
https://charts.kubesphere.io/main/ks-core-0.4.0.tgz \
--set internalTLS=true
```

## 5. 验证 KubeSphere 是否部署成功

a. 如果显示如下信息，则表明安装成功。

```
NOTES:
Please wait for several seconds for KubeSphere deployment to complete.

1. Make sure KubeSphere components are running:

    kubectl get pods -n kubesphere-system

2. Then you should be able to visit the console NodePort:

    Console: http://172.31.100.10:30880

3. To login to your KubeSphere console:

    Account: admin
    Password: "P@88w0rd"
    NOTE: Please change the default password after login.

For more details, please visit https://kubesphere.io.
```

b. 执行以下命令检查 Pod 状态。

```
root@ks01:~# kubectl -n kubesphere-system get pods
```

NAME	READY	STATUS	RESTARTS	AGE
ks-apiserver-5c86d9f4d7-59hdv	1/1	Running	0	41h
ks-console-7c46dd89db-vrrkm	1/1	Running	0	41h
ks-controller-manager-7b86d5f5f5-144m7	1/1	Running	0	2d19h

c. 当 Pod 状态都为 **Running** 时，从安装成功信息中的 Console、Account 和 Password 参数分别获取 KubeSphere 企业版 Web 控制台的 IP 地址、管理员用户名和管理员密码，并使用网页浏览器登录 KubeSphere 企业版 Web 控制台。

## KubeSphere Helm Chart 选项

### 常用选项

选项	默认值	描述
adminPassword	""	string - 为第一个管理员用户设置引导密码。登录后，管理员需要重置密码。如不设置，会使用内置默认密码 <b>P@88w0rd</b> 。
hostname	"example.com"	string - KubeSphere Server 完全限定的域名。
ingress.enabled	false	bool - 如果值为 true, 创建 KubeSphere Ingress 网关。
ingress.tls.enabled	true	bool - 如果值为 true, 则为 KubeSphere Ingress 网关启用 HTTPS。
ingress.tls.source	"generation"	string - Ingress 证书的来源，可选项: "generation, importation, letsEncrypt"。
letsEncrypt.email	""	string - 邮箱地址，证书过期时会收到邮件提醒。
letsEncrypt.environment	"production"	string - 可选项: "staging, production"。测试环境和生产环境证书在速率限制上存在区别，详情见: <a href="#">Let's Encrypt 官方文档</a> 。

高级选项

选项	默认值	描述
internalTLS	false	bool - 如果值为 true, 启用内部 TLS。console 和 apiserver 均会启用 HTTPS 服务。
ingress.ingressClassName	""	string - 使用网关的可选 Ingress 类, 可选项: "nginx, traefik"。
ingress.secretName	"kubesphere-tls-certs"	string - Ingress 网关所使用的包含 TLS 证书的 Secrets。
certmanager.duration	2160h	string - cert-manager 生成证书的过期时间。
certmanager.renewBefore	360h	string - cert-manager 刷新证书到证书过期之间的时间间隔。
global.imageRegistry	registry.cn-beijing.aliyuncs.com	string - 设置全局的 KubeSphere 镜像仓库地址。
global.tag	"v4.0.0"	string - 设置全局的 KubeSphere 镜像仓库标签。
apiserver.image.registry	""	string - 设置 ks-apiserver 镜像仓库地址。
apiserver.image.repository	"kse/ks-apiserver"	string - 设置 ks-apiserver 镜像名称。
apiserver.image.tag	""	string - 设置 ks-apiserver 镜像标签。
apiserver.nodePort	""	uint16 - 设置 ks-apiserver 服务 service 的 NodePort 端口。
console.image.registry	""	string - 设置 ks-console 镜像仓库地址。
console.image.repository	"kse/ks-console"	string - 设置 ks-console 镜像名称。
console.image.tag	""	string - 设置 ks-console 镜像标签。
console.nodePort	30880	uint16 - 设置 ks-console 服务 service 的 NodePort 端口。
controller.image.registry	""	string - 设置 ks-controller-manager 镜像仓库地址。
controller.image.repository	"kse/ks-controller-manager"	string - 设置 ks-controller-manager 镜像名称。
controller.image.tag	""	string - 设置 ks-controller-manager 镜像标签。

## 3. 卸载 KubeSphere 企业版

本节介绍如何卸载 KubeSphere Core，即卸载 KubeSphere 企业版。当前集群中运行的 Kubernetes 将不会被卸载。

### 警告

- 虽然此操作不会卸载 Kubernetes，如果当前集群中运行的业务使用了 KubeSphere 企业版提供的功能，此操作仍然可能导致业务中断。
- 此操作不能撤销，请谨慎执行此操作。

### 前提条件

为避免数据丢失，请提前备份所有重要数据。

### 操作步骤

执行以下命令卸载 KubeSphere 企业版。

```
helm -n kubespHERE-system uninstall ks-core
```