

# nginx-ingress 的安装使用

发表于 April 14, 2019

<https://unsplash.com/photos/fvaoHivSHGE>

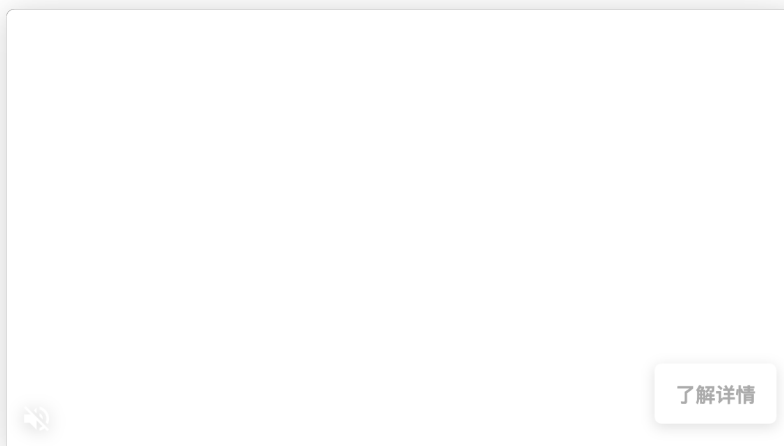
标签: [kubernetes](https://www.qikqiak.com/tags/kubernetes/) (<https://www.qikqiak.com/tags/kubernetes/>) [nginx](https://www.qikqiak.com/tags/nginx/) (<https://www.qikqiak.com/tags/nginx/>) [ingress](https://www.qikqiak.com/tags/ingress/) (<https://www.qikqiak.com/tags/ingress/>) [helm](https://www.qikqiak.com/tags/helm/) (<https://www.qikqiak.com/tags/helm/>) [traefik](https://www.qikqiak.com/tags/traefik/) (<https://www.qikqiak.com/tags/traefik/>)

nginx-ingress 和 traefik 都是比如热门的 ingress-controller，作为反向代理将外部流量导入集群内部，将 Kubernetes 内部的 Service 暴露给外部，在 Ingress 对象中通过域名匹配 Service，这样就可以直接通过域名访问到集群内部的服务了。相对于 traefik 来说，nginx-ingress 性能更加优秀，但是配置比 traefik 要稍微复杂一点，当然功能也要强大一些，支持的功能多许多，前面我们为大家介绍了 traefik 的使用，今天为大家介绍下 nginx-ingress 在 Kubernetes 中的安装使用。

## # 安装

我们这里通过 Helm 来简化 nginx-ingress 的安装，所以确保 Helm 能够正常使用，可以参考我们前面的文章：Kubernetes Helm 初体验。(<https://www.qikqiak.com/post/first-use-helm-on-kubernetes/>)

[i](#) [x](#)



由于 nginx-ingress 所在的节点需要能够访问外网，这样域名可以解析到这些节点上直接使用，所以需要让 nginx-ingress 绑定节点的 80 和 443 端口，所以我们这里通过 DaemonSet 和 hostPort 来进行部署，当然需要通过 nodeSelector 来筛选有外网 IP 的边缘节点。

```
$ kubectl get nodes --show-labels
NAME          STATUS    ROLES    AGE      VERSION   LABELS
ydzs-master   Ready     master   30d      v1.10.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=li
ydzs-node1    Ready     <none>   30d      v1.10.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=li
ydzs-node2    Ready     <none>   30d      v1.10.0   beta.kubernetes.io/arch=amd64,beta.kubernetes.io/os=li
```

比如我们集群使用 kubeadm 安装的，而且只有 master 节点有外网 IP，所以需要将 nginx-ingress 绑定在 master 节点上，通过 nodeSelector 绑定 label 标签：kubernetes.io/hostname=ydzs-master，当然还需要容忍该节点的污点，这个需要结合你的节点实际情况进行绑定，然后新建 custom.yaml 文件来覆盖 nginx-ingress Chart 包的一些默认参数：

```
controller:
  hostNetwork: true
  daemonset:
    useHostPort: false
    hostPorts:
      http: 80
      https: 443
  service:
    type: ClusterIP
  tolerations:
    - operator: "Exists"
  nodeSelector:
    kubernetes.io/hostname: ydzs-master

defaultBackend:
  tolerations:
    - operator: "Exists"
  nodeSelector:
    kubernetes.io/hostname: ydzs-master
```

要获取 Chart 包的默认参数值可以通过下面的命令获取：

```
$ helm fetch stable/nginx-ingress
$ tar -xvf nginx-ingress-1.4.0.tgz
```

nginx-ingress 目录下面的 values.yaml 文件即为默认参数值，大家可以根据自己的实际情况进行覆盖，然后使用下面的命令安装：

```
$ helm install stable/nginx-ingress --namespace kube-system --name nginx-ingress -f custom.yaml
```

Join Now: 50% Off 1 Year

广告 Get trusted insights & analysis you n  
informed and in control.

The Wall Street Journal

Open

安装完成后可以通过下面的命令查看 nginx-ingress 的 Pod 使用运行成功：

```
$ kubectl get pods -n kube-system | grep nginx-ingress
nginx-ingress-controller-587b4c68bf-vsqgm      1/1      Running    0      11h
nginx-ingress-default-backend-64fd9fd685-lmxhw  1/1      Running    0      1d
```

安装过程中需要用到 `gcr.io` 和 `quay.io` 的镜像，大家可以在 `dockerhub` 上搜索，pull 下来后重新打上 tag 即可。

## # 测试


上面的 `nginx-ingress` 安装成功后，我们可以通过一个简单的示例来测试下：(ngdemo.yaml)

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: my-nginx
spec:
  template:
    metadata:
      labels:
        app: my-nginx
    spec:
      containers:
        - name: my-nginx
          image: nginx:1.7.9
          ports:
            - containerPort: 80
---
apiVersion: v1
kind: Service
metadata:
  name: my-nginx
  labels:
    app: my-nginx
spec:
  ports:
    - port: 80
      protocol: TCP
      name: http
  selector:
    app: my-nginx
---
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: my-nginx
  annotations:
    kubernetes.io/ingress.class: "nginx"
spec:
  rules:
    - host: ngdemo.qikqiak.com
      http:
        paths:
          - path: /
            backend:
              serviceName: my-nginx
              servicePort: 80
```

直接创建上面的资源对象：


```
$ kubectl apply -f ngdemo.yaml
deployment.extensions "my-nginx" created
service "my-nginx" created
ingress.extensions "my-nginx" created
```

注意我们在 Ingress 资源对象中添加了一个 annotations: `kubernetes.io/ingress.class: "nginx"`, 这就是指定让这个 Ingress 通过 nginx-ingress 来处理。

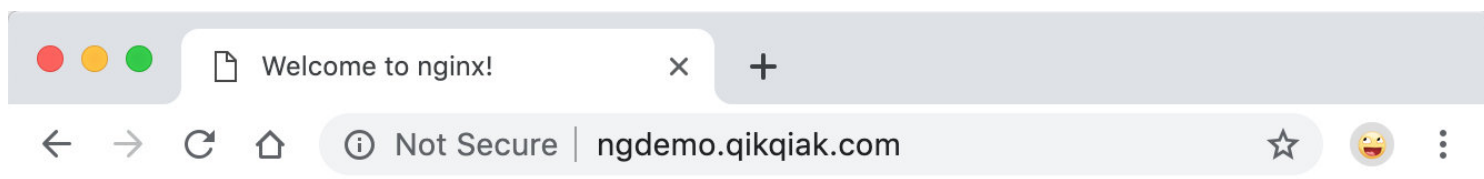


## Join Now: 50% Off 1 Year

In a time of ever-evolving complexity, our journalism keeps you informed and in control.



上面资源创建成功后, 然后我们可以将域名 `ngdemo.qikqiak.com` 解析到 `nginx-ingress` 所在的边缘节点中的任意一个, 当然也可以在本机 `/etc/hosts` 中添加对应的映射也可以, 然后就可以通过域名进行访问了。



# Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](https://nginx.org).  
Commercial support is available at [nginx.com](https://nginx.com).

*Thank you for using nginx.*

ngdemo