

Kubernetes Services

[Contact Sales \(/company...](#)[← Return to Glossary \(https://www.vmware.com/topics/glossary/\)](#)

What are Kubernetes Services?

A **Kubernetes service** is a logical abstraction for a deployed group of pods in a cluster (which all perform the same function).

Since pods are ephemeral, a service enables a group of pods, which provide specific functions (web services, image processing, etc.) to be assigned a name and unique IP address (clusterIP). As long as the service is running that IP address, it will not change. Services also define policies for their access.

Best Practices for Running Containers and Kubernetes in Production

Download now (https://tanzu.vmware.com/content/analyst-reports/best-practices-for-running-containers-and-kubernetes-in-production?src=WWW_us_ELQ_zc0xVZgdpMk3r8Hy9jqC&int_cid=7012H000001KPUip)

In Kubernetes, what is the difference between a service and a deployment?

In Kubernetes (<https://www.vmware.com/topics/glossary/content/kubernetes>), a deployment is a method of launching a pod with containerized applications and ensuring that the necessary number of replicas is always running on the cluster.

On the other hand, a service is responsible for exposing an interface to those pods, which enables network access from either within the cluster or between external processes and the service.

[Skip to Main Content](#)

vmware® (<https://www.vmware.com/>)



An Elevated View of the Tanzu Kubernetes Grid Service Architecture



What are the components of a Kubernetes services?

Kubernetes services connect a set of pods to an abstracted service name and IP address. Services provide discovery and routing between pods. For example, services connect an application front-end to its backend, each of which running in separate deployments in a cluster. Services use labels and selectors to match pods with other applications. The core attributes of a Kubernetes service are:

- A label selector that locates pods
- The clusterIP IP address and assigned port number
- Port definitions
- Optional mapping of incoming ports to a targetPort

Services can be defined without pod selectors. For example, to point a service to another service in a different namespace or cluster.

What are the types of Kubernetes services?

- **ClusterIP**. Exposes a service which is only accessible from within the cluster.
- **NodePort**. Exposes a service via a static port on each node's IP.
- **LoadBalancer**. Exposes the service via the cloud provider's load balancer.

• **ExternalName**. Maps a service to a predefined `externalName` field by returning a value for the CNAME record.

What is the Kubernetes ClusterIP service?

ClusterIP is the default type of service, which is used to expose a service on an IP address internal to the cluster. Access is only permitted from within the cluster.

What is a Kubernetes Headless Service?

Services that do not need load balancing and only expose a single IP can create a 'headless' service by specifying "none" as the clusterIP.

Headless services can be defined with selectors, in which case endpoint records are created in the API that modify the DNS to return addresses that point to pods that are exposing the service. Headless services without selectors don't create endpoint records. The DNS system configures either the CNAME record or a record for endpoints with the same name as the service.

What is a Kubernetes NodePort service?

NodePorts are open ports on every cluster node. Kubernetes will route traffic that comes into a NodePort to the service, even if the service is not running on that node. NodePort is intended as a foundation for other higher-level methods of ingress such as load balancers and are useful in development.

What is a Kubernetes ExternalName service?

ExternalName services are similar to other Kubernetes services; however, instead of being accessed via a clusterIP address, it returns a CNAME record with a value that is defined in the externalName: parameter when creating the service.

What is a Kubernetes Load Balancer service?

For clusters running on public cloud (<https://www.vmware.com/topics/glossary/content/public-cloud>) providers like AWS or Azure, creating a load LoadBalancer service provides an equivalent to a clusterIP service, extending it to an external load balancer that is specific to the cloud provider. Kubernetes will automatically create the load balancer, provide firewall rules if needed, and populate the service with the external IP address assigned by the cloud provider.

How do Kubernetes services work?

Services simply point to pods using labels. Since services are not node-specific, a service can point to a pod regardless of where it runs in the cluster at any given moment in time. By exposing a service IP address as well as a DNS service name, the application can be reached by either method as long as the service exists.

[Skip to Main Content](#)

How do you define a Kubernetes service?



vmware

(<https://www.vmware.com/>)



Services are defined in YAML, as are all Kubernetes objects. Suppose you deployed pods running a back-end service to process data coming from a web front end. To expose a service named 'service-backend' on the deployment 'deployment-backend' you would use:

```
apiVersion: v1
kind: Service
metadata:
  name: service-backend
spec:
  ports:
  - port: 4000
  protocol: TCP
  targetPort: 333
  selector:
  run: deployment-backend
  type: ClusterIP
```

The service 'service-backend' will be created, and any pod in the cluster can access it on their port 333 via (<http://service-backend:4000>)<http://service-backend:4000> (<http://service-backend:4000>), or at the cluster's IP address using port 4000.

Kubernetes services can also be created using the 'kubectl expose' command, which does not require a YAML file. The same service can be created using the command:

```
kubectl expose deployment deployment-backend --port=333 --target-port=4000 --name=service-backend
```

How do you access a Kubernetes service?

There are two ways to discover a Kubernetes service:

- **DNS (most common):** The DNS method is the recommended method of discovering services. To use this method, a DNS server must first be installed on the cluster. The DNS server monitors the Kubernetes API, and when a new service is created its name becomes available for easy resolution for requesting applications.
- **ENV variable:** This method relies on the kubelet adding environment variables for each active service for every node a pod is running on.

VMware Kubernetes Services related products, solutions, and resources

[Skip to Main Content](#)



vmware® (<https://www.vmware.com/>)



VMware Tanzu Kubernetes Grid

Discover how VMware Tanzu Kubernetes Grid makes it easy to install and run multi-cluster Kubernetes environments on any infrastructure.
(<https://tanzu.vmware.com/kubernetes-grid>)

Build containers from source code for Kubernetes

VMware Tanzu Build Service automates container creation, management, and governance at enterprise scale.
(<https://tanzu.vmware.com/build-service>)

VMware Tanzu Application Catalog

VMware Tanzu portfolio is a family of products and services for modernizing your applications and infrastructure to deliver better software to production.
(<https://tanzu.vmware.com/application-catalog>)

Transform your business - Why Tanzu

Master the art of building software your customers love—and use our portfolio of products to run it on any cloud.
(<https://tanzu.vmware.com/why-tanzu>)

Evolve Existing Apps with VMware Tanzu

VMware has a proven, pragmatic method to accelerate your app modernization. First, we help you adopt Kubernetes and container technologies.
(<https://www.vmware.com/cloud-solutions/app-modernization/existing-apps.html>)

Multi-cluster Kubernetes management

VMware Tanzu Mission Control is a centralized management platform for consistently operating and securing your Kubernetes infrastructure and modern applications across teams and clouds.
(<https://tanzu.vmware.com/mission-control>)

[Skip to Main Content](#)

Related Topics



vmware[®] (<https://www.vmware.com/>)



Related Topics

Kubernetes (<https://www.vmware.com/topics/glossary/content/kubernetes>)

Kubernetes Namespace (<https://www.vmware.com/topics/glossary/content/kubernetes-namespace>)

Kubernetes Networking (<https://www.vmware.com/topics/glossary/content/kubernetes-networking>)

Kubernetes Deployment (<https://www.vmware.com/topics/glossary/content/kubernetes-deployment>)

Kubernetes Architecture (<https://www.vmware.com/topics/glossary/content/kubernetes-architecture>)

Kubernetes Monitoring (<https://www.vmware.com/topics/glossary/content/kubernetes-monitoring>)

Kubernetes Pods (<https://www.vmware.com/topics/glossary/content/kubernetes-pods>)

Kubernetes Cluster (<https://www.vmware.com/topics/glossary/content/kubernetes-cluster>)

Kubernetes Infrastructure (<https://www.vmware.com/topics/glossary/content/kubernetes-infrastructure>)

Kubernetes Security (<https://www.vmware.com/topics/glossary/content/kubernetes-security>)

vmware[®] (<https://www.vmware.com/>)

Company

About Us (<https://www.vmware.com/company.html>)

Executive Leadership (<https://www.vmware.com/company/leadership.html>)

Newsroom (<https://www.vmware.com/company/news.html>)

Investor Relations (<https://ir.vmware.com>)

[Skip to Main Content](#)

Customer Stories (<https://www.vmware.com/company/customers.html>)



Diversity, Equity & Inclusion (<https://www.vmware.com/company/diversity.html>)

Environment, Social & Governance (<https://www.vmware.com/2030agenda.html>)



Careers (<https://careers.vmware.com/>)

Blogs (<https://blogs.vmware.com/>)

Communities (<https://communities.vmware.com/>)

Acquisitions (<https://www.vmware.com/company/acquisitions.html>)

Office Locations (https://www.vmware.com/company/office_locations.html)

VMware Cloud Trust Center (<https://cloud.vmware.com/trust-center>)

COVID-19 Resources (<https://www.vmware.com/company/news/updates/vmware-response-covid-19.html>)

Support

VMware Customer Connect (<https://www.vmware.com/go/customerconnect>)

Support Policies (<https://www.vmware.com/support/policies.html>)

Product Documentation (<https://docs.vmware.com>)

Compatibility Guide (<https://www.vmware.com/resources/compatibility/search.php>)

End User Terms & Conditions (<https://www.vmware.com/download/eula.html>)



Twitter (<https://twitter.com/VMware>)



YouTube (<https://www.youtube.com/user/vmwaretv>)



Facebook (<https://www.facebook.com/vmware>)



LinkedIn (<https://www.linkedin.com/company/vmware/mycompany/>)



Contact Sales (https://www.vmware.com/company/contact_sales.html)

[Skip to Main Content](#)



(<https://www.vmware.com/>)
© 2021 VMware, Inc.



Terms of Use (<https://www.vmware.com/help/legal.html>)

Your California Privacy Rights

(<https://www.vmware.com/help/privacy/california-privacy-rights.html>)

Privacy (<https://www.vmware.com/help/privacy.html>)

Accessibility (<https://www.vmware.com/help/accessibility.html>)

Site Map (https://www.vmware.com/site_index.html)

Trademarks (<https://www.vmware.com/help/trademarks.html>)

Glossary (<https://www.vmware.com/topics/glossary>)

Help (<https://www.vmware.com/help.html>)

Feedback