

METALLB

MetalLB is a load-balancer implementation for bare metal [Kubernetes](#) clusters, using standard routing protocols.

Note

Despite the beta status of the project / API, MetalLB is known to be stable and reliable.

The [project maturity](#) page explains what that implies.

Why?

Kubernetes does not offer an implementation of network load balancers ([Services of type LoadBalancer](#)) for bare-metal clusters. The implementations of network load balancers that Kubernetes does ship with are all glue code that calls out to various IaaS platforms (GCP, AWS, Azure...). If you're not running on a supported IaaS platform (GCP, AWS, Azure...), LoadBalancers will remain in the "pending" state indefinitely when created.

Bare-metal cluster operators are left with two lesser tools to bring user traffic into their clusters, "NodePort" and "externalIPs" services. Both of these options have significant downsides for production use, which makes bare-metal clusters second-class citizens in the Kubernetes ecosystem.

MetalLB aims to redress this imbalance by offering a network load balancer implementation that integrates with standard network equipment, so that external services on bare-metal clusters also "just work" as much as possible.

Requirements

MetalLB requires the following to function:

- A [Kubernetes](#) cluster, running Kubernetes 1.13.0 or later, that does not already have network load-balancing functionality.
- A [cluster network configuration](#) that can coexist with MetalLB.
- Some IPv4 addresses for MetalLB to hand out.
- When using the BGP operating mode, you will need one or more routers capable of speaking [BGP](#).
- When using the L2 operating mode, traffic on port 7946 (TCP & UDP, other port can be configured) must be allowed between nodes, as required by [memberlist](#).

Backward Compatibility

Previous versions of MetalLB are configurable via a `configmap`. However, starting from the version `v0.13.0`, it will be possible to configure it only via CRs. [A tool to convert](#) old configmaps to CRs is provided as a container image under `quay.io/metalb/configmaptocrs`.

Usage

The [concepts](#) section will give you a primer on what MetalLB does in your cluster. When you're ready to deploy to a Kubernetes cluster, head to the [installation](#) and [usage](#) guides.

FRR-K8s Mode

MetalLB uses [FRR-K8s](#) as the default backend for handling BGP sessions. FRR-K8s is a Kubernetes wrapper around [FRR](#) with its own API, allowing additional FRR configuration to be provided alongside MetalLB's while sharing the same BGP sessions.

It provides features that are not available with the native BGP implementation, such as pairing BGP sessions with BFD sessions and advertising IPv6 addresses.

An alternative [FRR mode](#) that configures FRR directly (without the FRR-K8s layer) is also available but **deprecated**. Please see the [installation](#) section for instructions on how to switch between modes.

Contributing

We welcome contributions in all forms. Please check out the [contributing guide](#) for more information.

One lightweight way you can contribute is to [tell us that you're using MetalLB](#), which will give us warm fuzzy feelings :).

We are a [Cloud Native Computing Foundation](#) sandbox project.


