

Peer finder

This is a simple peer finder daemon that is useful with StatefulSet and related use cases.

All it does is watch DNS for changes in the set of endpoints that are part of the governing service of the PetSet. It periodically looks up the SRV record of the DNS entry that corresponds to a Kubernetes Service which enumerates the set of peers for this the specified service.

Be sure to use the `service.alpha.kubernetes.io/tolerate-unready-endpoints` on the governing service of the StatefulSet so that all peers are listed in endpoints before any peers are started.

There are several ways to bundle it with your main application.

1. In an [init container](#), to help your pod determine its peers when it first started (determine the desired set of peers from the governing service of the StatefulSet. For this use case, the `--on-start` option can be used, but the `--on-change` option should not be used since the init container will no longer be running after the pod is started. An example of an `--on-start` script would be to edit a configuration file for the main app to insert the list of peers. This file needs to be on a Volume shared between the init container and the main container.
2. In a sidecar (e.g. a second container in the same pod as the main app), in which case the `--on-change` option can be used, but `--on-start` may not be useful without a way to guarantee the ordering of the sidecar relative to the main app container. An example of an on-change script would be to send an administrative command to the main container over the localhost network. (Note that signalling is not practical since pods currently do not share a PID namespace).
3. As pid 1 of the main container, in which case both `--on-change` and `--on-start` may be used. In this mode, the ordering of the peer-finder relative to the main app is ensured by having the peer finder start the main app. An example script would be to modify a configuration file and send SIGHUP to the main process.
4. Both 1 and 2.

Options 1 and 2 and 4 may be preferable since they do not require changes to the main container image. Option 3 is useful if signalling is necessary.

The peer-finder tool is intended to help legacy applications run in containers on Kubernetes. If possible, it may be preferable to modify an application to poll its own DNS to determine its peer set.

Not all StatefulSets are able to be scaled. For unscaleable StatefulSets, only the on-start message is needed, and so option 1 is a good choice.

DNS Considerations

Unless specified by the `-domain` argument, `peer-finder` will determine the FQDN of the pod by examining the `/etc/resolv.conf` file, looking for a `search` line and looking for the best match.

If your pod is not using the default `dnsPolicy` value which is `ClusterFirst` as the DNS policy, you may need to provide the `-domain` argument. In most common configurations, `-domain=cluster.local` will be the correct setting.