Diameter Endpoint

Feature Summary and Revision History

Summary Data

Table 1: Summary Data

Applicable Products or Functional Area	PCF
Applicable Platform(s)	SMI
Feature Default Setting	Enabled – Always-on
Related Documentation	Not Applicable

Revision History

Table 2: Revision History

Revision Details	Release
Enhancement introduced. PCF supports dual stack (IPv4 and IPv6) connectivity on its external interfaces/endpoints.	2022.01.0
Enhancement introduced. PCF supports IPv6 connectivity on its external interfaces/endpoints.	2021.04.0
First introduced.	Pre 2020.01.0

Feature Description

You can enable the Diameter endpoint to dynamically create pods on a designated node or host. This feature might be a requirement when you want to ensure that the nodes are meeting specific security and regulatory parameters, or the node is closer to the datacenter in terms of geographical proximity. The node affinity determines the node where PCF creates the Diameter endpoint pods, which are based on the affinity towards a node or group of nodes. Node affinity is a set of rules that allows you to define the custom labels on nodes and specify the label selectors within the pods. Based on these rules, the scheduler determines the location where the pod can be placed.

Note If you do not specify a node, then the Kubernetes scheduler determines the node where the Diameter endpoint creates a pod.

PCF supports both IPv4 and IPv6 connectivity on its external interfaces/endpoints (inbound and outbound).

Configuring the Node for the Diameter Endpoint Pod

This section describes how to specify the node or host where the Diameter endpoint must spawn the pod.

Note Configuration changes to the diameter endpoint cause the endpoint to restart automatically. Cisco recommends making such changes only within the maintenance window.

To specify the node where you want Diameter endpoint to spawn the pod, use the following configuration:

```
config
   diameter group diameter_group_name
   mode server server_name
   stack stack_name
     application application_name
     bind-ip ipv4 host_address
     bind-ipv6 ipv6 host_address
     bind-port port_number
     fqdn fqdn_address
     realm realm_address
     node-host node_host_address
     end
```

NOTES:

- **diameter group** *diameter_group_name*—Specify the Diameter group name.
- **mode server** *server_name*—Specify the server name that operates as the mode server.
- **stack** *stack_name*—Specify the stack name.
- **application** *application_name*—Specify the application name.

- **bind-ip** *host_address*—Specify the host address IPv4 to bind the stack.
- **bind-ipv6** *host_address*—Specify the host address IPv6 to bind the stack.
- **bind-port** *port_number*—Specify the port number to bind the stack.
- **fqdn** *fqdn_address*—Specify the FQDN address.
- **realm** *realm_address*—Specify the realm address.
- **node-host** *node_host_address*—Specify the host IP address of the node.

Sample Configuration

The following is a sample configuration of the node configuration.

```
mode server
  stack cicdsite
  application rx
  bind-ip 192.0.2.18
  realm cisco.com
  node-host for-node-2a-worker39e1587354h
  exit
```