

Diameter Endpoint

- Feature Summary and Revision History, on page 1
- Feature Description, on page 2
- Configuring the Node for the Diameter Endpoint Pod, on page 2
- **RX-AAR messages handling during site failover**, on page 3

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
```

RX-AAR messages handling during site failover

Table 3: Feature History

Feature Name	Release Information	Description
Cross-Site Session Handling for Distributed PCF	2025.04.0	This feature enables seamless cross-site session handling within a distributed Policy Control Function (PCF) system. The feature enables PCF nodes to process Rx session-related messages for sessions created at other sites. When the system is configured to accept Diameter messages with unknown destination hostnames, session processing continues during site failovers, load balancing, or dynamic routing scenarios. **Command introduced:** **diameter properties**

Cross-site session handling in distributed PCF

Cross-Site Session Handling in a Distributed Policy Control Function (PCF) system allows PCF nodes to accept and process Diameter messages even when the Destination-Host field in the message does not match

the hostname of the receiving PCF node. This capability is crucial for maintaining session continuity and processing Rx session-related messages across different PCF sites in a distributed architecture, particularly during site failover events or when Session Border Gateways (SBGs) dynamically route traffic.

The key capabilities are:

- Cross-Site message processing: Enables PCF nodes to process Diameter messages related to existing sessions, regardless of which PCF site initially created the session.
- Seamless failover handling: Facilitates uninterrupted processing of session-related messages during site outages, routing changes, or load balancing, ensuring session continuity and reliability.
- Bypasses strict Hostname validation: When enabled, the Diameter protocol stack on the PCF node bypasses its strict validation of the Destination-Host field, allowing messages with unknown hostnames to be accepted and processed.

Configuring diameter properties for Cross-site session handling

This procedure describes how to enable PCF nodes to accept Diameter messages with unknown Destination-Host values, which is essential for cross-site session handling in a distributed PCF environment.

Before you begin

Ensure you meet these important prerequisites before you start with the Ops Center configuration:

- Ensure you have administrative access to the system's Configuration Management Framework.
- Understand the impact of this configuration change on your distributed PCF system.

Follow these steps to configure Diameter properties for cross-site session handling.

Procedure

Step 1 Enter the Ops Center to specify the Diameter properties.

Example:

```
pcf(config)# diameter properties
```

Step 2 Change the `jdiameter.accept.unknown_desthost` property value from false to True.

a) **false**: The Diameter protocol stack rejects messages where the Destination-Host does not match the hostname of the receiving PCF node, returning error code 3002 (DIAMETER_UNKNOWN_PEER).

b) **true**: The Diameter protocol stack accepts messages with unknown Destination-Host values and bypasses strict hostname validation. This allows the PCF to process the message locally or retrieve session state.

Example:

```
pcf(config)# diameter properties jdiameter.accept.unknown_desthost value true
Tue Sep  9  06:21:31.289 UTC+00:00
pcf(config-properties-jdiameter.accept.unknown_desthost)# commit
Tue Sep  9  06:21:34.615 UTC+00:00
```

After applying the changes, verify that the `jdiameter.accept.unknown_desthost` property is successfully set to `true` on the PCF nodes. You can verify the property by querying the configuration or by observing system behavior. Applying this property config redeploys the diameter pods.