IP Address Management

Feature Summary and Revision History

Summary Data

Table 1: Summary Data

Applicable Product(s) or Functional Area	cnBNG
Applicable Platform(s)	SMI
Feature Default Setting	Disabled - Configuration Required
Related Changes in this Release	Not Applicable
Related Documentation	Not Applicable

Revision History

Table 2: Revision History

Revision Details	Release
First introduced.	2021.01.0

Feature Description

IP Address Management (IPAM) is a method of tracking and managing IP addresses of a network. IPAM is one of the core components of the subscriber management system. Traditional IPAM functionalities are insufficient in Cloud-Native network deployments. Hence, IPAM requires additional functionalities to work

with the Cloud-Native subscriber management system. The Cloud-Native IPAM system is used in various network functions, such as Session Management function (SMF), Policy Charging function (PCF), and Broadband Network Gateway (BNG).

The IPAM system includes the following functionalities to serve the Cloud Native and Control and User Plane Separation (CUPS) architecture:

- **Centralized IP Resource Management**—Based on the needs of the Internet Service Provider (ISP), the Control Plane (CP) is deployed either on a single (centralized) cluster or multiple (distributed) clusters. For multiple cluster deployments, the IPAM automatically manages the single IP address space across the multiple CPs that are deployed in the distributed environment.
- **IP Address-Range Reservation per User Plane**—For subscribers connecting to the Internet core, the User Plane (UP) provides the physical connectivity. The UP uses the summary-routes to advertise subscriber routes to the Internet core. For CPs that are managing multiple UPs, the CP reserves a converged IP subnet to the UPs. In such a scenario, the IPAM splits the available address space into smaller address-ranges and assigns it to different UPs.
- **IP Address Assignment from Pre-Reserved Address-Ranges**—When subscribers request for an IP address, the IPAM assigns addresses from the pre-reserved address range of their respective UP.

IPAM Components

This section describes the different components of the IPAM system.

IPAM Sub-Modules

The IPAM functionalities are categorized in the following sub-modules:

IPAM Server

This module manages the complete list of pools and address-space configurations. It splits the configured address-ranges into smaller address-ranges (statically or dynamically) to distribute it to the IPAM Cache modules. The IPAM server can be deployed as a centralized entity to serve a group of CN clusters or as an integrated entity within a single cluster.

IPAM Cache

This module acquires the free address-ranges from the IPAM server and allocates individual IP addresses to the IPAM clients. The IPAM cache is generally deployed in the Distributed mode running within each cluster, to communicate with the co-located or remotely located IPAM server. It is also responsible for address-range reservation per UP and pool threshold monitoring. The IPAM server and cache modules can also run in an integrated mode.

IPAM Client

This module is tightly coupled with its respective network-function, responsible for handling request and release of individual IP address from the IPAM cache for each IP managed end-device.

Unlike the IPAM server and cache module, the IPAM client caters to use-cases specific to network-functions such as BNG, SMF, PCF, and so on.

IPAM Integration in cnBNG

The Cloud-Native Broadband Network Gateway (cnBNG) function comprises of loosely coupled microservices that provide the functionality of the BNG. The decomposition of these microservices is based on the following three-layered architecture:

1. Layer 1: Protocol and Load Balancer Services (Stateless)
2. Layer 2: Application services (Stateless)
3. Layer 3: Database Services (Stateful)

The IPAM and cnBNG integration occurs in the Application Services layer.

BNG Node Manager Application—The BNG Node Manager application is responsible for the User Plane function (UPF) management, ID and resource management, and IP address management. Therefore, the IPAM Cache is integrated as part of this microservice.

Also, the UPF uses the IPAM Client module for address-range-reservation per UPF.

BNG DHCP and PPPOE Application—The BNG-DHCP and BNG-PPOE pods are responsible for providing IP addresses to the BNG subscriber session. During session bring-up, the IP address is requested and during session bring-down, the IP address is released back. These First Sign of Life (FSOL) applications send the inter-process communications (IPC) to the Resource Manager (RMGR) component in the NodeMgr. The NodeMgr receives the IPC and invokes the IPAM component.

IPAM Server Application—Based on the deployment model, the IPAM Server runs as an independent microservice as part of the same cluster or in a remote cluster.

In standalone deployments, the IPAM Server functionality is an integral part of the IPAM Cache, that is, it runs as part of the Node Manager microservice itself.

How it Works

This section describes the call flow pertaining to the integration of the IPAM in the cnBNG.

Call Flows

This section describes the following IPAM call flows in cnBNG:

- IPAM initial sequence call flow
- IPAM call flow
- IPAM static-pool call flow

IPAM Initial Sequence Call Flow

This section describes the cnBNG initial sequence call-flow.

Figure 1: IPAM Initial Sequence Call Flow

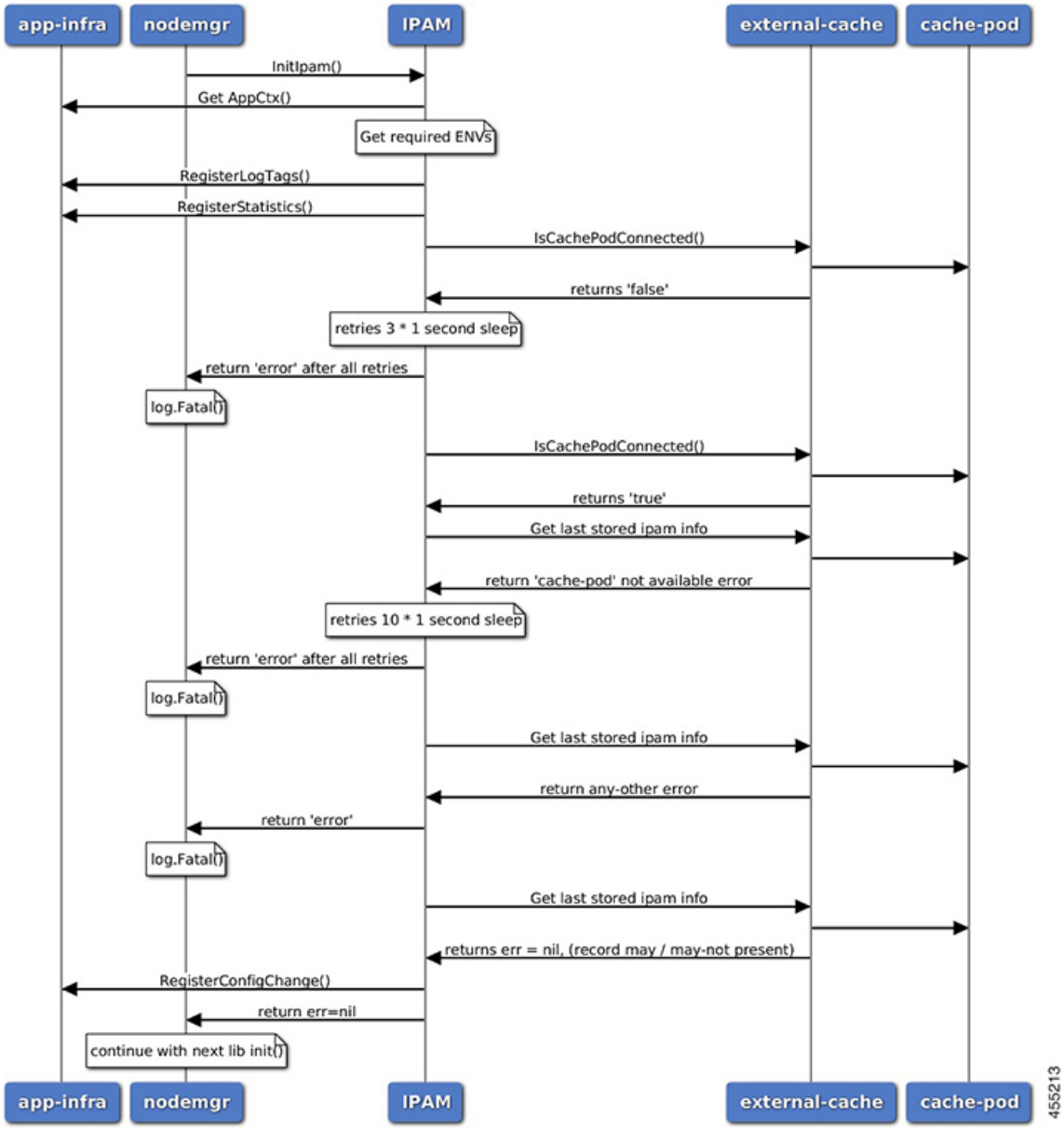

Table 3: IPAM Initial Sequence Call Flow Description

Step	Description
1	IPAM reads the required environments, registers with the application infrastructure for log-tags, metrics, and database connection.
2	IPAM restores the previous state from the cache-pod, if present.
3	IPAM registers for configuration change and applies the new configuration change, if any. -change, apply new config-changes if any

IPAM Call Flow

This section describes the cnBNG IPAM call-flow.

Figure 2: IPAM Call Flow

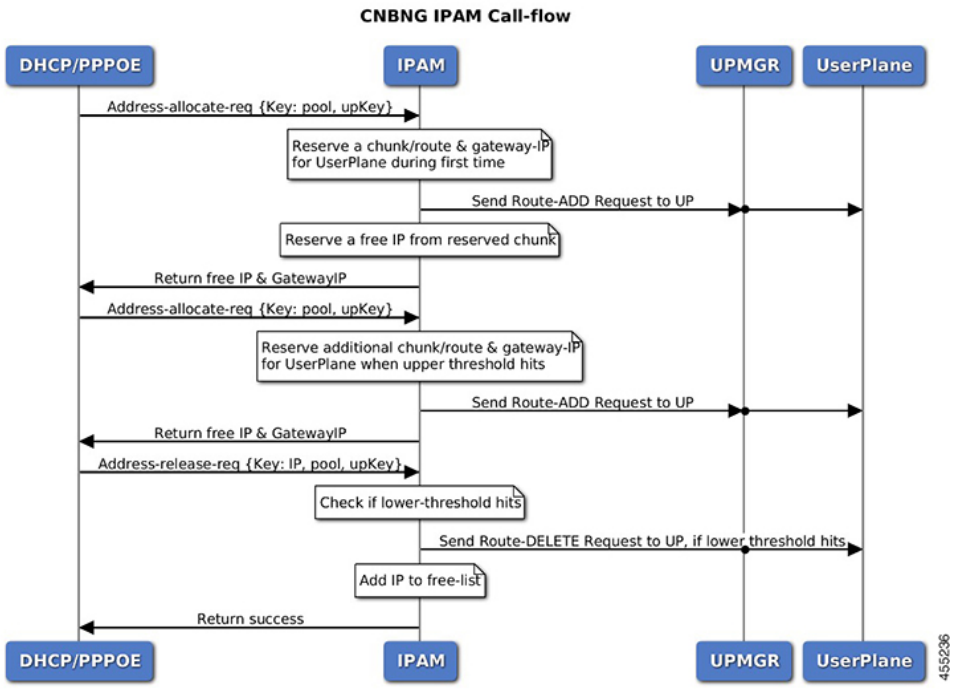

Table 4: IPAM Call Flow Description

Step	Description
1	IPAM receives the ‘addr-alloc’ request from the DHCP or PPPoE pod with pool-name, addr-type and user plane function (UPF) as input.
2	IPAM reserves a new address-range (if not already present for UPF) and sends a ROUTE-ADD message to the UPF. It waits for a success or failure response. If the receives a failure response, it removes the chunk and repeats this step.
3	IPAM reserves a free-IP from the assigned address-range and returns to the DHCP or PPPoE.
4	IPAM monitors the ‘upper-threshold’ for each UPF during each IP address-allocation and also has a background thread that monitors. It then assigns new address-ranges to the UPF and repeats the ROUTE-ADD flow.
5	IPAM receives the ‘addr-free’ request from the DHCP or PPPoE pod with pool-name, addr-type, addr or pfx, and UPF as input.
6	IPAM moves the addr or pfx first to the quarantine-list until the quarantine timer and later moves it to the free-list.

Step	Description
7	IPAM monitors the ‘lower-threshold’ (currently 0%) of the address-range of each UPF, removes the address-range from the UPF, and sends the ROUTE-DELETE message.

IPAM Static-Pool Call Flow

This section describes the IPAM static-pool call flow.

Figure 3: IPAM Static Pool Call Flow

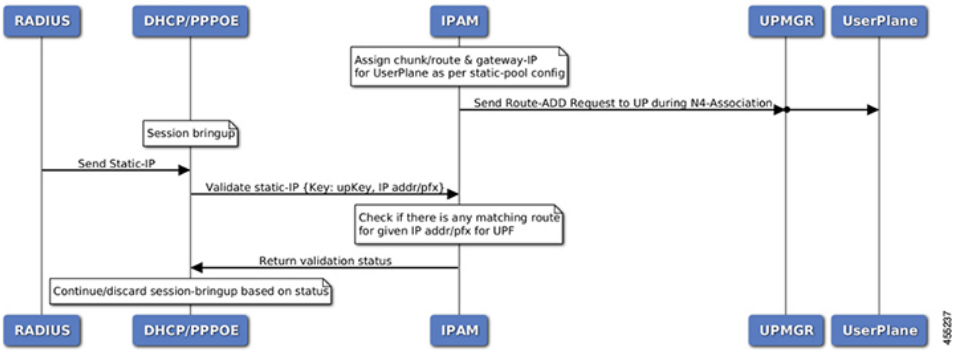

Table 5: IPAM Call Flow Description

Step	Description
1	IPAM receives the ‘addr-alloc’ request from the DHCP or PPPoE pod with pool-name, addr-type and user plane function (UPF) as input.
2	IPAM reserves a new address-range (if not already present for UPF) and sends a ROUTE-ADD message to the UPF. It waits for a success or failure response. If the receives a failure response, it removes the chunk and repeats this step.
3	IPAM reserves a free-IP from the assigned address-range and returns to the DHCP or PPPoE.
4	IPAM monitors the ‘upper-threshold’ for each UPF during each IP address-allocation and also has a background thread that monitors. It then assigns new address-ranges to the UPF and repeats the ROUTE-ADD flow.
5	IPAM receives the ‘addr-free’ request from the DHCP or PPPoE pod with pool-name, addr-type, addr or pfx, and UPF as input.
6	IPAM moves the addr or pfx first to the quarantine-list until the quarantine timer and later moves it to the free-list.
7	IPAM monitors the ‘lower-threshold’ (currently 0%) of the address-range of each UPF, removes the address-range from the UPF, and sends the ROUTE-DELETE message.

Limitations

The IPAM feature has the following limitations:

- Duplicate IP address is not supported within a pool.
- Duplicate IP address is not supported across pools, that belong to same VRF.
- Removal of ‘pool’ is not supported while addresses are already assigned.
- Removal or modification of IP-address-ranges is not supported while addresses are already assigned.
- Change of ‘source’ field is not supported while address or prefixes are already assigned.
- Change of ‘vrf-name’ of pool is not supported while address or prefixes are already assigned.
- Start-address should be less than the End-address.
- Configuring addr-range split-size in wrong manner, that is, size of address-range < size-of-per-cache < size-of-dp, is not supported.
- Configuring IPv6 Address (IANA) and Prefix (IAPD) values interchangeably is not supported.
- Configuring invalid ‘prefix-length’ for Prefix (IAPD) range is not supported.

Configuring IPAM Feature

This section describes how to configure the IPAM feature.

Configuring the IPAM feature involves the following steps:

1. Configuring IPAM source
2. Configuring the global threshold
3. Configure IPAM address pool
4. Configuring IPv4 address ranges
5. Configuring IPv6 address ranges
6. Configuring IPv6 prefix ranges
7. Configuring the IPv4 threshold
8. Configuring the IPv6 threshold
9. Configuring IPv4 address range split
10. Configuring IPv6 address and prefix address-range split

Configuring IPAM Source

Use the following configuration to configure the IPAM source.

```
config
  ipam
```

```
    source local
      threshold { ipv4-add percentage | ipv6-address percentage | ipv6-prefix
percentage }
       commit
```

NOTES:

- **ipam**: Enters the IPAM Configuration mode.
- **source local**: Enters the local datastore as the pool source.
- **threshold { ipv4-add** *percentage* **| ipv4-address** *percentage* **| ipv6-prefix** *percentage* **}**: Specifies the threshold in percentage for the following:
  - **ipv4-add** *percentage*: Specifies the IPv4 threshold. The valid values range from 1 to 100. The default value is 80.
  - **ipv6-add** *percentage*: Specifies the IPv4 threshold. The valid values range from 1 to 100. The default value is 80.
  - **ipv6-prefix** *percentage*: Specifies the IPv6 threshold prefix. The valid values range from 1 to 100. The default value is 80.

Configuring Global Threshold

Use the following configuration to configure the global threshold.

```
config
   ipam
      threshold
         ipv4-addr percentage
         ipv6-addr percentage
         ipv6-prefix percentage
         commit
```

NOTES:

- **ipam**: Enters the IPAM Configuration mode.
- **threshold**: Enters the threshold sub-mode.
- **ipv4-add** *percentage*: Specifies the IPv4 threshold. The valid values range from 1 to 100. The default value is 80.
- **ipv6-add** *percentage*: Specifies the IPv4 threshold. The valid values range from 1 to 100. The default value is 80.
- **ipv6-prefix** *percentage*: Specifies the IPv6 threshold prefix. The valid values range from 1 to 100. The default value is 80.

Configuring IPAM Address Pool

Use the following configuration to configure the IPAM address pool.

```
config
   ipam
     address-pool pool_name [ address-quarantine-timer ] [offline ] [ static
 user_plane_name ] [ vrf-name string ]
      commit
```

NOTES:

- **ipam**: Enters the IPAM configuration mode.
- **address-pool** *pool_name* **[address-quarantine-timer] [offline] [static** *user_plane_name* **] [vrf-name** *string* **]**: Configures the address pool configuration. *pool_name* must be the name of the address pool.

  This command configures the following parameters:

  - **offline**: Sets the address pool to offline mode.
  - **static** *user_plane_name*: Specifies the 'user-plane' name associated to this static-pool.
  - **vrf-name** *string*: Configures the Virtual routing and forwarding (VRF) name of the pool.

Configuring IPv4 Address Ranges

Use the following configuration to configure the IPv4 address ranges.

```
config
   ipam
      address-pool pool_name
         ipv4
            address-range start_ipv4_address end_ipv4_address [ default-gateway
ipv4_address ] [ offline ]
            commit
```

NOTES:

- **ipam**: Enters the IPAM configuration mode.
- **address-pool** *pool_name*: Configures the address pool configuration. *pool_name* must be the name of the address pool.
- **ipv4**: Enters the IPv4 mode of the pool.
- **address-range** *start_ipv4_address end_ipv4_address* **[default-gateway** *ipv4_address* **] [offline]**: Configures the IPv4 address range with the starting and ending IPv4 address.
  - **default-gateway** *ipv4_address*: Specifies the IPv4 address of the default gateway.
  - **offline**: Sets the address pool to offline mode.

Configuring IPv6 Address Ranges

Use the following configuration to configure the IPv6 address ranges:

```
config
   ipam
```

```
address-pool pool_name
   ipv6
      address-range start_ipv6_address end_ipv6_address [ offline ]
      commit
```

NOTES:

- **ipam**: Enters the IPAM configuration mode.
- **address-pool** *pool_name*: Configures the address pool configuration. *pool_name* must be the name of the address pool.
- **ipv6**: Enters the IPv6 mode of the pool.
- **address-range** *start_ipv6_address end_ipv6_address* **[offline]**: Configures the IPv6 address range with the starting and ending IPv6 address.

  [offline]: Sets the address pool to offline mode.

Configuring IPv6 Prefix Ranges

Use the following configuration to configure the IPv6 prefix ranges:

```
config
   ipam
      address-pool pool_name
         ipv6
            prefix-ranges
               prefix-range prefix_value prefix-length prefix_length
               commit
```

NOTES:

- **ipam**: Enters the IPAM configuration mode.
- **address-pool***pool_name*: Configures the address pool configuration. *pool_name* must be the name of the address pool.
- **ipv6**: Enters the IPv6 mode of the pool.
- **prefix-ranges**: Enters the prefix ranges mode.
- **prefix-range** *prefix_value* **prefix-length** *length*: Configures the IPv6 prefix range. *prefix_value* specifies the IPv6 prefix range.

  prefix-length *length* specifies the IPv6 prefix length.

Configuring IPv4 Threshold

Use the following configuration to configure the IPv4 threshold:

```
config
   ipam
      address-pool  pool_name
         ipv4
            threshold
```

```
upper-threshold percentage
commit
```

NOTES:

- **ipam**: Enters the IPAM Configuration mode.
- **address-pool** *pool_name*: Configures the address pool configuration. *pool_name* must be the name of the address pool.
- **ipv4**: Enters the IPv4 mode of the pool.
- **threshold**: Enters the threshold sub-mode.
- **upper-threshold** *percentage*: Specifies the IPv4 upper threshold value in percentage. The valid values range from 1 to 100. The default value is 80.

The following is a sample configuration:

```
config
   ipam
      address-pool p1
         ipv4
            threshold
               upper-threshold 80
```

Configuring IPv6 Prefix-Range Threshold

Use the following configuration to configure the IPv6 prefix-range threshold.

```
config
   ipam
      address-pool pool_name
         ipv6
            prefix-ranges
               threshold
                  upper-threshold percentage
                  commit
```

NOTES:

- **ipam**: Enters the IPAM configuration mode.
- **address-pool** *pool_name*: Configures the address pool configuration. *pool_name* must be the name of the address pool.
- **ipv6**: Enters the IPv6 mode of the pool.
- **prefix-ranges**: Enters the IPv6 prefix ranges sub-mode.
- **threshold**: Enters the threshold sub-mode.
- **upper-threshold** *percentage*: Specifies the IPv6 upper-threshold value in percentage.

The following is an example configuration:

```
config
   ipam
      address-pool p3
         ipv6
```

```
      prefix-ranges
         threshold
            upper-threshold 78
```

Configuring IPv4 Address Range Split

Use the following configuration to configure the IPv4 address range split.

```
config
   ipam
      address-pool pool_name
         ipv4
            [ no ] split-size { per-cache value | per-dp value }
            commit
```

NOTES:

- **ipam**: Enters the IPAM configuration mode.
- -**address-pool** *pool_name*: Configures the address pool configuration. *pool_name* must be the name of the address pool.
- **ipv4**: Enters the IPv4 mode of the pool.
- **[no] split-size { per-cache** *value* **| per-dp** *value* **}**: Specifies the size of the IPv4 range to be split for each IPAM cache allocation. The IPAM server consumes this configuration. The **no** form of this command disables the splitting of the address-ranges into smaller chunks.

  per-cache *value*: Specifies the size of the IPv4 range to be split for each Data-Plane (User-Plane) allocation. The valid values range from 2 to 262144. The default value is 1024.

  The IPAM cache consumes this configuration.
- **per-dp** *value*: Specifies the size of the IPv4 range to be split for each Data-Plane (User-Plane) allocation. The valid values range from 2 to 262144 The default value is 256.

  The IPAM cache consumes this configuration.

Configuring IPv6 Address and Prefix Address-Range-Spilt

Use the following configuration to configure the IPv6 address and prefix address range spilt.

```
config
   ipam
      address-pool pool_name
         ipv6
            address-ranges
               [ no ] spilt-size { per-cache value | per-dp value }
               commit
            prefix-ranges
               [ no ] spilt-size { per-cache value | per-dp value }
               commit
```

NOTES:

- **ipam**: Enters the IPAM configuration mode.

- **address-pool** *pool_name*: Configures the address pool. *pool_name* must be the name of the address pool.
- **ipv6**: Enters the IPv6 mode of the pool.
- **[no] spilt-size { per-cache** *value* **| per-dp** *value* **}**: Specifies the size of the IPv6 range to be split for each IPAM cache allocation. The IPAM server consumes this configuration. The **no** form of this command disables the splitting of the address-ranges into smaller chunks.

  per-cache *value*: Specifies the size of the IPv6 range to be spilt for each Data-Plane (User-Plane) allocation. The valid values range from 2 to 262144. The default value is 1024.

  The IPAM cache consumes this configuration.
- **per-dp** *value*: Specifies the size of the IPv6 range to be spilt for each Data-Plane (User-Plane) allocation. The valid values range from 2 to 262144 The default value is 256.

  The IPAM cache consumes this configuration.