

IP Pool Planning Guidelines

- IP Distribution in CUPS Architecture, on page 1
- UP Group Concept, on page 1
- When to Add New Pool, on page 2
- IP Pool Fine-tuning Parameters, on page 4
- Dynamic IP Pool Planning Guidelines, on page 5
- Static IP Pool Guidelines, on page 7
- Implications of Taking Very Big Chunk Size, on page 7

IP Distribution in CUPS Architecture

If IP Pool is configured in UP, it becomes tightly coupled to the UP. If the IP pool is unused to a large percentage, it becomes a waste of resources. UPs, which are short of IP resources could benefit from the unused resources, if they are available to them. Similarly, if one UP becomes unreachable, the IP range allocated to it can't be reused. To overcome this limitation, IP chunking mechanism is introduced.

In CUPS architecture, IP pools are configured in the CP, and CP is responsible for the IP Pool management. Any IP pool configured on the CP is divided into chunks of configured size. During the UP-registration process, the CP identifies all the APNs which are being served by that particular UP and the associated IP pool configurations in each APN. The CP allocates the chunks from these IP pools to the UP. CP monitors the chunk usage at per pool level for each UP every **chunk-threshold-timer** second, and if the chunks threshold is reached, it allocates new chunks to the UP. Similarly, if certain IP chunks in the UP aren't utilized, and if CP doesn't have enough free chunks in reserve for that pool, CP withdraws those IP chunk resources from the respective UPs.

In CUPS architecture, IP pools aren't considered as resource/criteria for UP selection. To avoid uneven load distribution across UPs, the operator must ensure that sufficient IP chunks are available for all APNs and UPs in an UP Group.

UP Group Concept

When the UP associates with CP, chunks from IP pool associated with APN are distributed to the UP. UP in turn is associated with single UP Group, which is tied to APN. So, the chunks from IP pool that are configured as part of APN are distributed to the UPs belonging to the UP group associated with that APN. Therefore, the size of UP group must be kept in consideration while planning the pool size and chunk size.

Note UP can only be part of single UP Group. You must avoid adding an UP to two UP groups, because it can cause undefined behavior.

Default UP Group

When an APN doesn't have UP Group configured explicitly, they are considered part of default UP group. A default UP Group is used mostly for consumer traffic where the APNs are too large catering to large number of subscribers and operating on large IP pool.

Specific UP Group

To direct traffic of specific APN to selective set of UPs, specific UP Group is used. For example, consider a case of enterprise subscribers where APN and IP pool are small (order of size 256). In this case, it's better not to chunk the pool and dedicate the entire APN to the UP group consisting of only one UP. Similarly, consider very large deployments consisting of large APNs and 20+ UPs. Even in this case, there's no large APN that it requires all 20 UPs to serve it. To better utilize chunking mechanism of IP pools, divide these UPs and APNs into some specific UP groups.

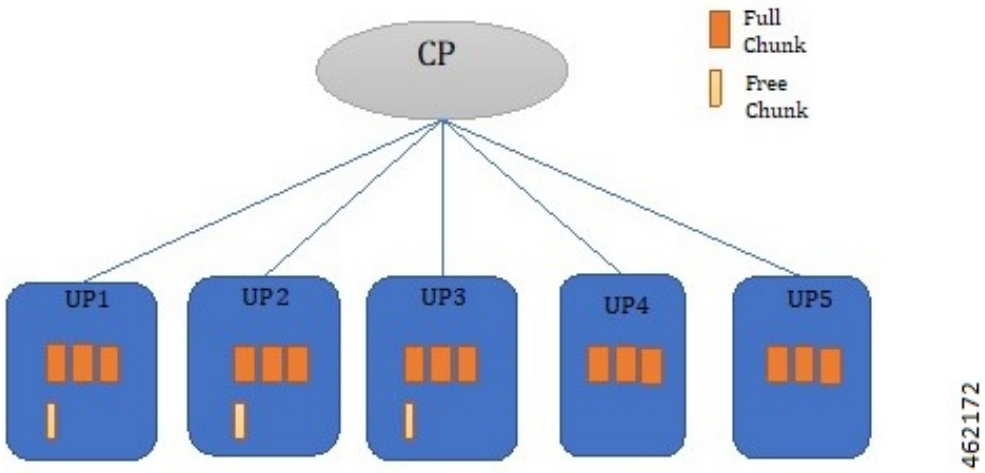
For example, consider a case where you have 20 UPs and require the UPs to serve 100k subscribers each. Consider that there are 25 APNs each having a pool of size 16 K with chunk size of 1k, and six APNs of size 64k, and chunk size of 1k. In this case, to utilize IP pools better, you can create two UP Groups, UP group-1 with four UPs to cater 25 small APNs and UP group-2 consisting of 16 UPs and six large APNs.

When to Add New Pool

When the amount of free chunks left for an APN are less than the number of UPs in the UP Group, and if the selected UP has all chunks fully utilized (that is, no free chunks left), then it leads to uneven load distribution. Also, UP selection algorithms are overridden and IP is given from the UP, which has a chunk available.

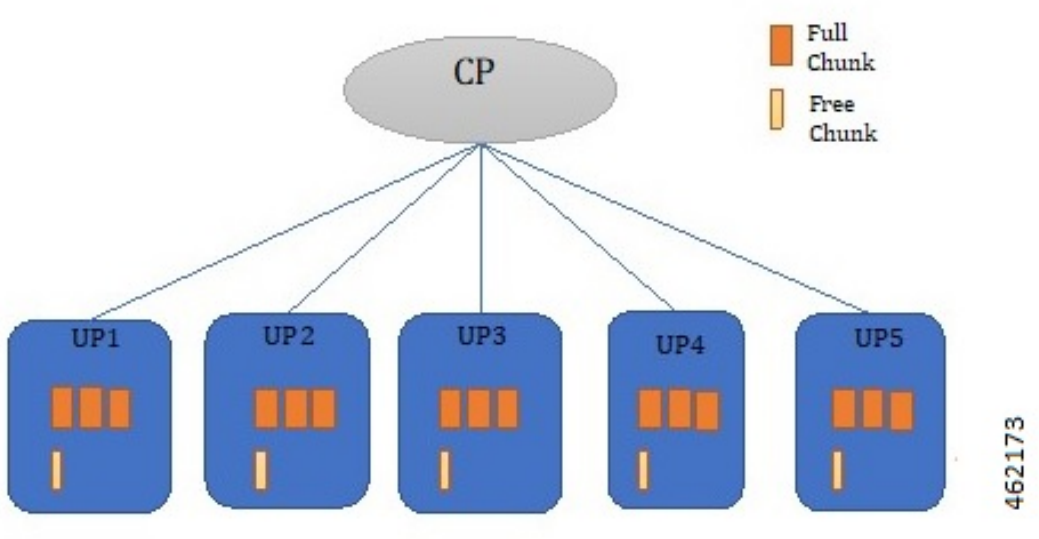
This scenario is illustrated in the following figure:

Figure 1: When to Add New Pool



Add more pools in that APN to avoid uneven load distribution, and chunking should be done to make chunks as even as possible.

Figure 2: Scenario After the New Pool is Added



Note We recommend you to plan the dimensioning in advance for future use, and add more chunks to avoid uneven load distribution.

IP Pool Fine-tuning Parameters

Threshold Timer

CP pushes or removes chunk to/from all UPs periodically. The periodicity is configured with the help of **chunk-threshold-timer**. If the UP has used more than 70% of address allocated from a pool, then a new chunk is pushed to the UP. Similarly, If the UP is underutilizing IP pool resources, then CP pulls back unused chunk from the UP. Removal of chunk is also dependent on other factors described in the section Chunk Withdrawal.

Chunk Withdrawal

CP withdraws chunk from UPs which are underutilizing that pool when CP has free chunks less than **min-chunks-threshold-per-pool** at periodic interval. If the UP is using less than 40 % of allocated IP address from the pool, and if the UP has more than two free chunks available, then a chunk is withdrawn from it.

Initial Chunk Pushed

Initial chunk pushed to each UP can be controlled using **cups max-user-plane** keyword, and it works at context level. Initial chunks are pushed depending on the following factors:

- Total free chunks in pool (command **show ip pool** displays this value)
- Total chunks in pool (value configured in **cups max-user-planes** CLI). By default, the value is set to 10 for max-user-planes.
- Three chunks

Example 1:

The CUPS **max-user-plane** CLI command is used to decide how many initial chunks are pushed to the UP when it registers. The number of initial chunks pushed depends on the following factors:

- If chunks < **max-user-plane** value, then one chunk is pushed.
- If chunks are more, then it's minimum of (chunks/**max-user-plane**, 3).

  Default value of max-user-plane is 10.

Consider that you have eight chunk pools, and want to use four UPs in deployment. You may be interested to give all chunks directly to all UPs. You can do the same by setting **max-user-plane** as 4. Similarly, if you want to have eight UPs in deployment, you can configure the value as 8, and only one chunk is pushed to an UP.

Moreover, it provides some extra chunks which can serve as buffer to cater to sudden surge of subscribers. This surge of subscribers might exceed the normal threshold chunk replenishment rate. Consider that there's one pool with 32 chunks and four UPs of 2k size, and incoming subscribers rate is 2k per minute. If you want to have extra buffer chunk, that is, guard chunk to avoid load imbalance due to incoming subscribers being higher than chunk replenishment rate in small window, then you can use **max-user-plane** CLI. If you don't want to use this CLI, then you can set it to 1000, and always one chunk is pushed.

Chunk Size

Chunk size should be planned considering both CEPS and evenness of chunk with respect to number of UPs in UP Group. Configuring too large chunks can lead to unevenness among UPs with respect to IP addresses. But, if the chunk is too small, then chunk replenishment rate might be lower than incoming subscribers. Low chunk replenishment rate leads to UP override or load imbalance.

Dynamic IP Pool Planning Guidelines

Chunking Guidelines

Chunking should be planned in consideration with the UP-Group scale. There are associated implications of a too large or too small sized chunk, so maintaining a good balance is important.

- Smaller chunk size reduces the imbalance between the chunk distribution among the UPs. But, very small chunk sizes lead to faster chunk exhaustion at the UP, and negatively impacts the CEPS rate.
- Very large chunk size may result in uneven distribution of chunks among the UPs. This issue may lead to UP override or load imbalance on certain UPs where no IPs are available, while IP address is still available with another UP.

The following shows a sample configuration where better IP pool resource planning is recommended:

- **IPv6 pool:** Poolv6_example_1 pool_group_example1 x:x:x:x::/48 chunk_size = 8192
- **IPv6 pool:** Poolv6_example_1 pool_group_example1 x:x:x:x::/48 chunk_size = 8192
- **UPs associated with APN:** UP1, UP2, UP3, UP4, and UP5
- **Threshold timer:** 60 secs
- **Incoming Subscriber rate per UP:** 6k subscribers per minute

(Considering that only one IP pool group is attached to the APN, which this UP group is serving)

Now, both IP pools have 65536 IP addresses.

Case of chunk size being too large Consider that addresses are divided into eight chunks to be distributed among five UPs. This means that not all UPs can get the comparable number of IP resources, and some UPs exhaust their IP resources sooner than the others.

Once IP resources reach exhaustion, UP override or load imbalance happens on such UPs. In this example, it can start happening after the usage of around 40960 addresses per pool or around 81920 addresses at IP pool group level.

- Pool1: (**UP1** = 8192 + 8192, **UP2** = 8192 + 8192, **UP3** = 8192 + 8192, **UP4**=8192, **UP5**=8192)
- Pool2: (**UP1** = 8192 + 8192, **UP2** = 8192 + 8192, **UP3** = 8192 + 8192, **UP4**=8192, **UP5**=8192)

Case of chunk size being too small: Consider same pools as described in the previous section are designed with chunk size of 512 with 128 chunks. In that case,

- Pool1: (**UP1** = 26 *512, **UP2** =26*512, **UP3** = 26*512, **UP4**=25*512, **UP5**=25*512)
- Pool2: (**UP1** = 26 *512, **UP2** =26*512, **UP3** = 26*512, **UP4**=25*512, **UP5**=25*512)

Although in this case, UP override or load imbalance due to chunk imbalance happens after 128000 (50 chunks*512 size*5 UPs) addresses is used, each UP has only 1k address per minute from each UP. Since the threshold timer is 60 sec, but incoming subscriber rate is 6k, each UP sees load imbalance for 5k subscribers.

Case of Correct design: Better design is to divide the IP pool into chunk size, for example, 4096 IP addresses. This design provides the CP with 16 chunks to distribute to five UPs.

Pool1: (**UP1** = 4096 + 4096 + 4096 + 4096, **UP2** = 4096 + 4096 + 4096, **UP3** = 4096 + 4096 + 4096, **UP4**= 4096 + 4096 + 4096, **UP5**= 4096 + 4096 + 4096)

In this case, UP override or load imbalance due to chunk imbalance happens after 122880 (six chunks *4096 size*5 UPs) addresses is used, and each UP has only 8k addresses per minute from each UP. Since threshold timer is 60 sec and incoming subscriber rate is 6k, UPs can cater to all subscribers.

As evident in this case, chunk size of 4096 results in better IP resource distribution among the UPs than having 8096 chunk size. As the IP resources reach exhaustion, the UP override or load imbalance happens on such UPs. In this example, it starts happening after the usage of around 61440 addresses per pool. Also, because there are two IP pools in the pool group at the same time, chunk replenishment is 4k from pool1 + 4k from pool2, which is greater than the required rate of 6k.

UP Grouping Guidelines

You must separate the consumer and enterprise customers in different UP groups since the IP pool size and routing requirements for each of them are different. Default UP group is used for the consumer customers. It's recommended to create specific User Groups for the enterprise customers and associate with enterprise APNs. Though the chunking mechanism provides the efficient IP address management for larger pools, it should be avoided in case of smaller pool sizes (for example, IP pool size less than 4k). It's recommended to dedicate the smaller IP pools to a particular UP. You can achieve this by having single UP in specific UP group and associating it to the APN.

Consider a case where you have five UPs and want to serve 40 small enterprise APNs of 256 addresses, and eight large consumer APNs of 64k addresses each. In this case, make two UP-groups, UPGroup1 consisting of one UP and 40 small enterprise APNs, and UPGroup2 consisting of four UPs and eight large consumer APNs.

UP Addition Guidelines

Before adding a new UP in a UP group, operator should ensure that there are free chunks available for all APNs at CP, which can be allocated during UP registration. This ensures that there are no uneven load distribution, as this UP still gets selected for call distribution even if one APN doesn't have any IP address available for this UP.

Miscellaneous Guidelines

- Use Pool Usage threshold alarms to get warning about the replenishment of IP pool resources. Take appropriate action, that is, adding a new pool in pool group.
- For IPv4v6 sessions in case of dynamic v4v6 address allocation, both IPv4 and IPv6 addresses that need to be assigned to UE belong to the same UP. IP pool planning must ensure that enough v4 and v6 IP addresses are available at per UP level on which v4v6 calls are expected.

Consider one IPv4 pool with name “poolv4” x.x.x.x/17 of 32k address and IPv6 pool with name “poolv6” x:x:x:x:/48 of 64k address. Consider that there are two APNs, APN1 which wants to make v4v6 call and operates on both “poolv4” and “poolv6”:

ip pool poolv4 209.165.200.224/17 – 32k address used by APN1

ipv6 pool poolv6 prefix 2003::/48 - 64k address used by APN2

If APN2 ends up using more than 32k IPv6 address, then there might not be sufficient IPv6 addresses left for v4v6 for APN1 due to poor planning of APNs. Segregate poolv6 of 64k address into poolv6_1 of 32k address used by APN1 and poolv6_2 of 32k address used by APN2. If needed, APN2 should add more IPv6 pools if it requires more IP addresses rather than starving APN1 of its fair share.

Static IP Pool Guidelines

Static IP pool is used in following scenarios:

- UE sends IP Address in Initial Attach.
- AAA/S6b returned IP Address.
- DHCP returned IP Address.

In the case of static IP pool, address is already decided by UE, and the benefit of UP selection goes away. Only the UP that has the chunk, which contains the selected IP address, can serve that call. For static IP pool, chunks are given to UP when UE requests first IP from an unused chunk. These chunks are given to UPs in a UP group in round robin fashion. Static chunks once allocated are never taken back from UP except in the case of Sx restart. The following are the guidelines with respect to static pools:

- For static IP pool, make one call as part of MOP procedure to dedicate that chunk to a particular UP. This helps in avoiding latency on first call, as the chunk is pushed on first call only.
- For static IP pool, use non-default UP group with limited number of UPs serving all the APNs with static IP pool.
- For static IP pool, use uniform chunk size across all pools. Pools should be of uniform size to avoid load imbalance on UPs, as UP selection can’t be used with static IP pool as explained before.
- For the static IPv4v6 PDNs to be successful, both IPv4 and IPv6 addresses need to be on the same UP. Only way to ensure this is to have a single UP in the UP group.
- For the multi-PDNs on same APN with one PDN static and other PDN dynamic to be successful, both addresses must be on the same UP. Since for dynamic pool, address is selected by UP selection algorithm, and for static address, UP is decided by the IP selected, the only way to avoid load imbalance is to have single UP in the UP group.

Implications of Taking Very Big Chunk Size

Pool System Limits

Currently CP DI-Large model supports the scaling numbers for parameters as listed in the table given below. These limits remain constant irrespective of the chunk size value that is used and represent the maximum

allowed limit for any given parameter. The limits of the parameter which have reached their maximum value restricted the subsequent parameter's upper limit value.

Note Small and medium model comparatively has lower limits.

Parameters	Limits
IPv4 pools per context	2000
IPv6 pools per context	256
IP pools per chassis	5000 (including both v4 and v6)
Dynamic pool addresses	16 Million per context 32 Million per chassis
Static pool addresses	32 Million per context 96 Million per chassis
Number of VRFs	300 per context 2048 per chassis
Max IP Pool size	512k
Max IPv6 Pool size	1 Million

Implications of chunk size on UP group:

The pool is the basic unit for chunk allocation and all UPs are allocated chunks from relevant pools. Maximum UPs which can get chunks with chunk size value of 65536 are One Million/65536 = 16. Due to which only 16 UPs are supported in each UP group for the chunk size value being 65536.

Implications of chunk size on APN:

For a single UP group used in an APN configuration, the limits are the same as the UP group limit values.

For multiple UP groups used in an APN configuration, refer to the *Multiple UP Groups with Group Specific IP Pool* chapter. The maximum UP groups of 16 UPs that are supported are 16 Million addresses per context or One Million address pool allowing a total of 16 UP groups of 16 UP APN.

Due to the exhaustion of all pool configuration in the v6 pool, the rest of the APN operating in the same VPN context uses the same IPv6 pool. The 16 UP groups of 16 UPs are based on the assumption that there are no IPv4 addresses as otherwise the limit is lower than expected. The system supports around 32 Million dynamic addresses, while only two SGI contexts are allowed.