ACOS 6.0.8
IPv4-to-IPv6 Transition Solutions Guide

December, 2025

PATENT PROTECTION

A10 Networks, Inc. products are protected by patents in the U.S. and elsewhere. The following website is provided to satisfy the virtual patent marking provisions of various jurisdictions including the virtual patent marking provisions of the America Invents Act. A10 Networks, Inc. products, including all Thunder Series products, are protected by one or more of U.S. patents and patents pending listed at:

a10-virtual-patent-marking.

TRADEMARKS

A10 Networks, Inc. trademarks are listed at: a10-trademarks

CONFIDENTIALITY

This document contains confidential materials proprietary to A10 Networks, Inc.. This document and information and ideas herein may not be disclosed, copied, reproduced or distributed to anyone outside A10 Networks, Inc. without prior written consent of A10 Networks, Inc.

DISCLAIMER

This document does not create any express or implied warranty about A10 Networks, Inc. or about its products or services, including but not limited to fitness for a particular use and non-infringement. A10 Networks, Inc. has made reasonable efforts to verify that the information contained herein is accurate, but A10 Networks, Inc. assumes no responsibility for its use. All information is provided "as-is." The product specifications and features described in this publication are based on the latest information available; however, specifications are subject to change without notice, and certain features may not be available upon initial product release. Contact A10 Networks, Inc. for current information regarding its products or services. A10 Networks, Inc. products and services are subject to A10 Networks, Inc. standard terms and conditions.

ENVIRONMENTAL CONSIDERATIONS

Some electronic components may possibly contain dangerous substances. For information on specific component types, please contact the manufacturer of that component. Always consult local authorities for regulations regarding proper disposal of electronic components in your area.

FURTHER INFORMATION

For additional information about A10 products, terms and conditions of delivery, and pricing, contact your nearest A10 Networks, Inc. location, which can be found by visiting www.a10networks.com.

Table of Contents

Large Scale Network Address Translation

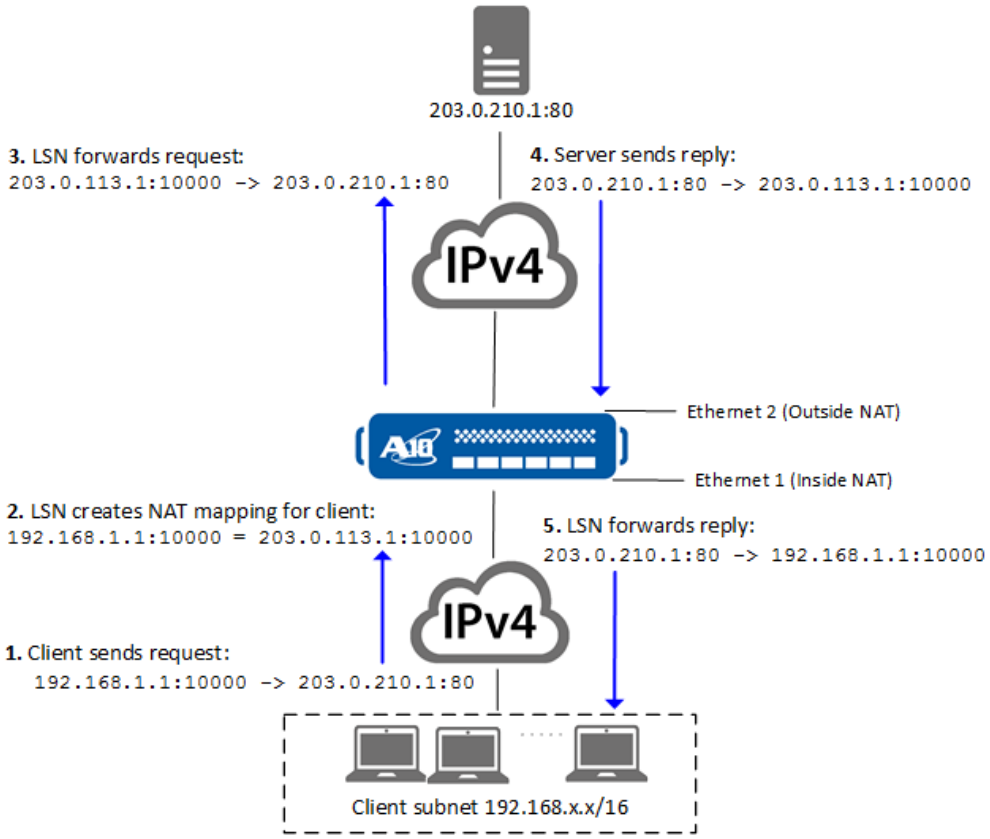
This chapter describes how Large Scale Network Address Translation (LSN) works and how to configure it.

The following topics are covered:

Overview

LSN provides robust NAT support for network carriers, also known as Internet Service Providers. Carriers can use LSN to provide NAT service for multiple enterprises and residential clients. Figure 1 illustrates an example of a carrier that uses LSN to provide NAT to residential clients.

Figure 1 : Large Scale NAT



The carrier's clients are on an internal subnet, 192.168.1.x/24, in the carrier's network. When a client sends a request, ACOS creates a mapping of the client's internal address and protocol port to a public address and protocol port. In this example, LSN creates the following mapping:

- Client Internal Address: 192.168.1.1:10000
- Client Public Address: 203.0.113.1:10000

After LSN creates an IP address mapping for a client, it uses the same mapping for all traffic between the client and an external IP address by default. In this example, the client 192.160.1.1:1000 assigns the same NAT IP and port to the following destinations:

Table 1 : Same NAT IP Assigned to All Destination IP Addresses

Source IP Address	Destination IP Address	NAT IP Address
192.168.1.1:10000	203.0.210.1	203.0.113.1:10000
192.168.1.1:10000	203.0.220.1	203.0.113.1:10000
192.168.1.1:10000	203.0.230.1	203.0.113.1:10000

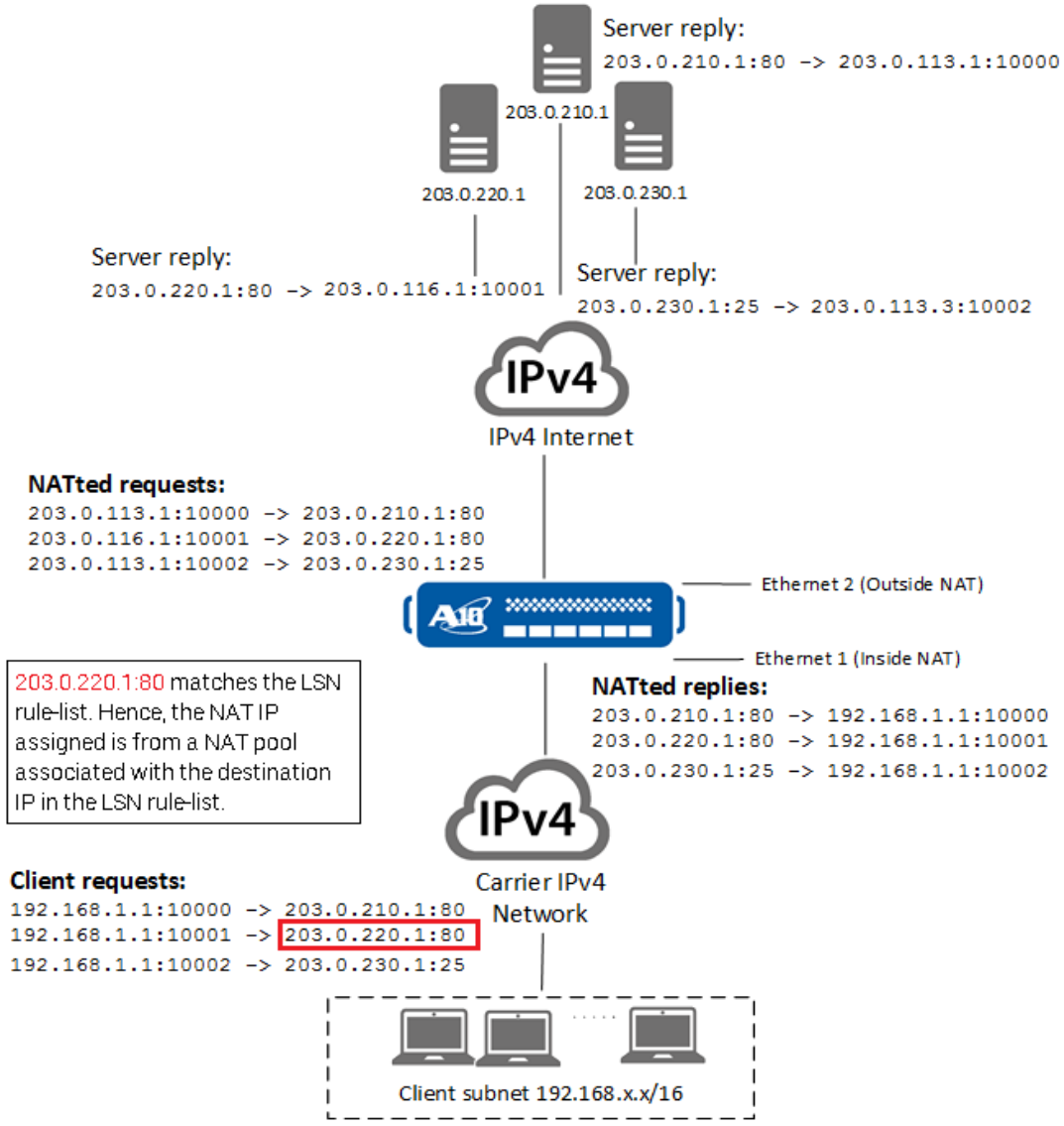
However, the default sticky NAT behavior can be overridden using the LSN rule-list configuration. An LSN rule-list lets you allocate a NAT pool for the configured destination IP address. If a destination IP address is configured in the LSN rule-list, then the NAT IP is assigned from the NAT pool allocated in the rule-list.

In this example, the client 192.160.1.1:1000 opens multiple sessions. Assume that the destination IP 203.0.220.1:80 is configured in the LSN rule-list. LSN assigns the NAT IP 203.0.116.1:10000 from the NAT pool allocated in the rule-list. In such a scenario, LSN would create the following mapping (see Figure 2):

Table 2 : Different NAT IP Assigned to Configured Destination IP Address

Source IP Address	Destination IP Address	NAT IP Address
192.168.1.1:10000	203.0.210.1	203.0.113.1:10000
192.168.1.1:10001	203.0.220.1	*203.0.116.1:10000*
192.168.1.1:10002	203.0.230.1	203.0.113.1:10000

Figure 2 : LSN Address Mapping



The implementation of LSN conforms to the following RFCs:

- draft-nishitani-cgn-02 – This is the main RFC for LSN
- RFC 4787 – Network Address Translation (NAT) Behavioral Requirements for Unicast UDP
- RFC 5382 – NAT Behavioral Requirements for TCP
- RFC 5508 – NAT Behavioral Requirements for ICMP

Remember the following issues:

- Carrier Grade NAT (CGN) and LSN are interchangeable terms.
- LSN, NAT64/DNS64, and DS-Lite can be used on the same ACOS device.
- LSN, NAT64, and DS-Lite can use the same NAT pool.

For information about Port Control Protocol (PCP), see Port Control Protocol for LSN. For information about logging, see the Traffic Logging Guide for IPv6 Migration.

Comparing LSN and Traditional NAT

Unlike NAT, LSN offers the following options:

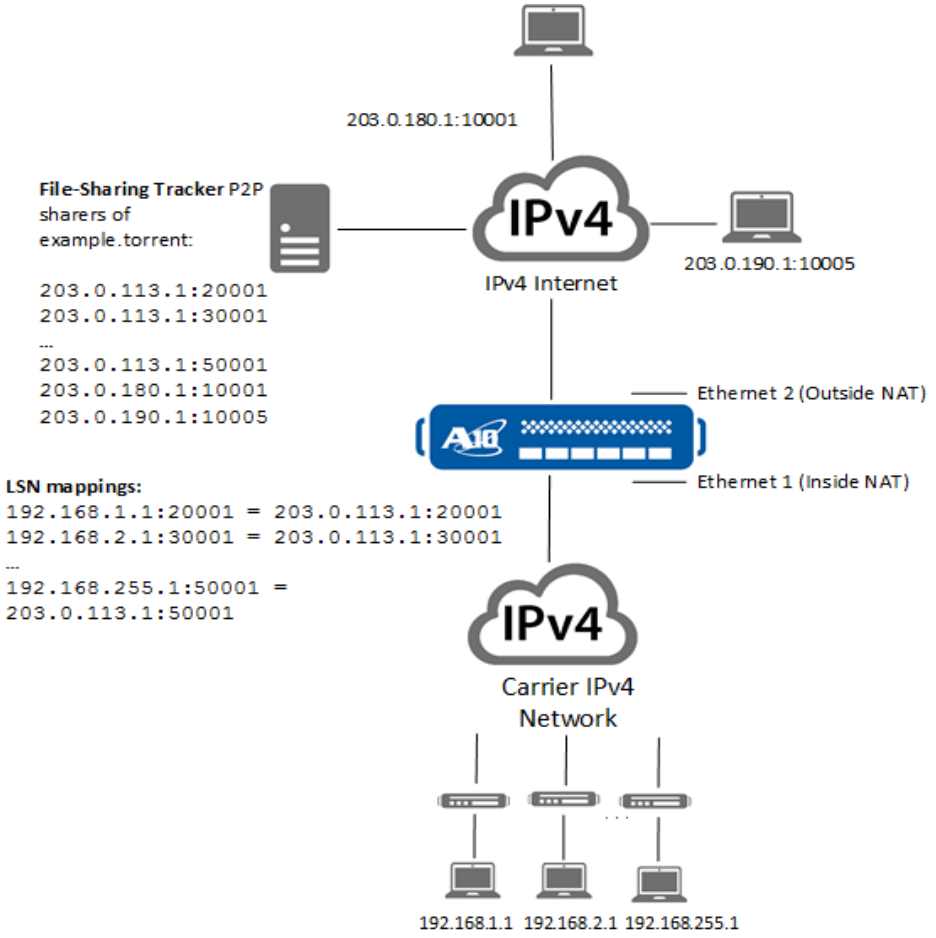
- High transparency – Existing user applications continue to work with minimal to no impact on customers.
- Well defined NAT behavior – LSN's consistent, deterministic behavior allows for easy development of new user applications. Traditional NAT implementations vary considerably, and may not work for all applications.
- Fairness in resource sharing – LSN provides user guarantees and protection.
- LSN works for both client-server (traditional) and client-client (P2P) applications.

Traditional NAT works for client-to-server applications, where a client opens a connection to a server and requests data, and the server responds back to the client. However, traditional NAT is often inadequate for contemporary applications such as peer-to-peer (P2P) file-sharing, instant messengers (IM), and Voice-over-IP (VoIP).

NOTE: To provide NAT for these types of applications, LSN is required.

Figure 3 shows an example of P2P file sharing among LSN clients and other devices.

Figure 3 : LSN Clients Using P2P File Sharing



In this example, multiple clients are registered with a P2P file-sharing tracker as sharers of the `example.torrent` file. All clients are registered on the file-sharing tracker by their public IP addresses. LSN allows each of the internal clients to use the same public IP address, with different Layer 4 source port numbers. LSN also allows the clients in the internal subnet to share the file between clients and with other clients that are outside the internal network.

NOTE: When possible, LSN uses the internal client's source protocol port number in the external mapping for the client. However, if the protocol port is already used by another client on the same external IP address, LSN selects another protocol port for the new mapping.

Sticky NAT

Sticky NAT enables a client to use the same NAT IP address in a NAT pool for all destinations, and that is the default behavior. When all user sessions are cleared, then a different NAT IP may be assigned.

For example, the client 192.168.1.1:10000 opens multiple HTTP sessions to connect to the following servers:

- 203.0.210.0
- 203.0.220.0
- 203.0.230.0

When the client 192.168.1.1 sends a request, ACOS creates a mapping of the client's internal address and protocol port to a public address and protocol port. After LSN establishes an IP address mapping for a client, it uses the same mapping for all traffic between the client and the destination IP addresses.

Table 3 : Same NAT IP Addresses Assigned to Destination IP Addresses

Source IP Address	Destination IP Address	NAT IP Address
192.168.1.1:10000	203.0.210.1	203.0.113.1:10000
192.168.1.1:10001	203.0.220.1	203.0.113.1:10000
192.168.1.1:10002	203.0.230.1	203.0.113.1:10000

Some applications that open multiple sessions to the same or multiple servers often work better with sticky NAT.

LSN can override the sticky NAT behavior by using the LSN rule-list. For more information, refer to Destination Based NAT IP Address.

To adhere to strictly sticky NAT behavior regardless of configuring destination IP addresses in the LSN rule-list, use the following command:

```
ACOS(config)# cgnv6 lsn strictly-sticky-nat
```

Destination Based NAT IP Address

LSN can override the default sticky NAT behavior for certain destinations configured on the LSN rule-list. An LSN rule-list lets you allocate a NAT pool for the configured destination IP address. If a destination IP address is configured in the LSN rule-list, then the NAT IP is assigned from the NAT pool allocated in the rule-list.

By allocating NAT pool based on the destination IP addresses, ACOS provides more efficient, scalable, and flexible NAT pool management.

When a client IP is associated with more than one NAT IP addresses, multiple user-quota sessions are created. Each session can be associated with different NAT IP addresses.

For performing source NAT using an allocated pool for a destination IP address, refer to Configuring the Rule-list.

Full-Cone NAT

To overcome the shortcomings of traditional NAT, LSN implements full-cone NAT, also known as one-to-one NAT, has the following behaviors:

- Endpoint-Independent Mapping (EIM) – After LSN maps an internal client's source IP address and Layer 4 (TCP or UDP) port to an external IP address and port, the same mapping is used for all traffic from that internal source IP and port, regardless of the destination. However, multiple endpoint-independent mappings (EIMs) can exist for the same client's source IP address and protocol port if multiple NAT IPs are mapped using the LSN rule-list configuration.
- If a client is mapped to multiple NAT IPs due to LSN rule-list configuration, then

  For pings, the ICMP query identifier is treated the same way as a UDP or TCP port:

  - Internal-IP-and-L4-Port = External-IP-and-L4-Port for all destinations
  - Internal-IP-and-ICMP-query-ID = External-IP-and-ICMP-query-ID for all destinations

- Endpoint-Independent Filtering (EIF) – For traffic from any source to a mapped client, LSN always allows the traffic to be forwarded to the internal client regardless of the endpoint. For an example of this behavior, see Figure 3.

These techniques provide consistent NAT mapping behavior, which enables client-to-client applications such as P2P, client-to-server applications, and NAT traversal techniques such as STUN, to work correctly.

Consider the following information:

- EIF is different from security filtering that is provided by ACLs, black/white lists, and so on.

  With EIF, LSN does not cause an internal client to be unreachable by certain sources and by using different mappings that are based on destination. The ACOS device's security features can still be used to control access to clients.

- EIM must be enabled in order to enable EIF.

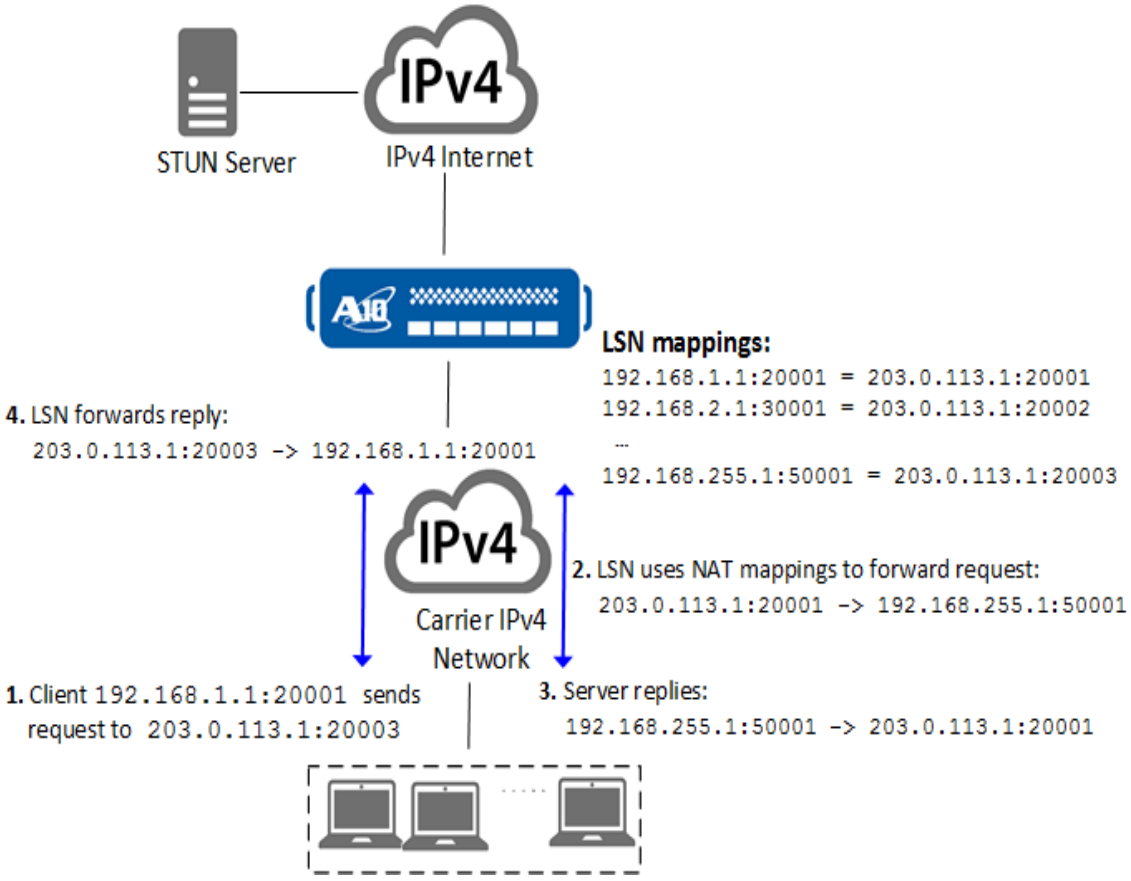
NOTE: The EIM/EIF behavior is the same for all NAT44, NAT64, DS-Lite, and Fixed-NAT.

Hairpinning

Hairpinning allows inside clients to communicate by using the clients' outside addresses and is useful for applications that require global addresses.

Figure 4 illustrates an example of hairpinning.

Figure 4 : LSN NAT – Hairpinning



LSN filters traffic to prevent self hairpinning, which occurs when the traffic that is initiated by an inside client is routed back to itself.

Hairpinning Filtering

You can change hairpin filtering to one of different levels of granularity:

- **Self-IP**—Self-IP hairpin filtering drops hairpin traffic from a client to its own NAT address, regardless of the source protocol port. For example, inside client 10.10.10.10:10000 is mapped to public address 203.0.113.1:10000. The traffic that client 10.10.10.10 sends to destination address 203.0.113.1:10000 is dropped.

The traffic is dropped even if the source protocol port is different from the port number used in the client's mapping. For example, traffic from 10.10.10.10.8000 and from 10.10.10.10:10000 is dropped.

NOTE: When enabled, self-IP hairpin filtering applies to both TCP and UDP traffic.

- **Self-IP-Port**—Self-IP-port hairpin filtering drops traffic when the following conditions are met:
  - The destination is the client's own public IP address
  - The source IP address and protocol port are the address and port that are used in the client's NAT mapping.

    The option is useful where double NAT is used, because more than one client might be behind a single NAT IP address. For example, clients 10.10.10.x:3000 and 10.10.10.x:4000 are both behind 192.168.1.1:10 and 192.168.1.1:20, respectively. From LSN's perspective, 192.168.1.1 is one inside client. LSN creates a single mapping for the first traffic from 192.168.1.1, and uses that mapping for subsequent traffic from any client behind 192.168.1.1. If the first traffic comes from 192.168.1.1:3000, LSN creates a mapping to 203.0.113.1:3000. Subsequent traffic from 192.168.1.1:3000 or 192.168.1.1:4000 uses mapping 203.0.113.1:3000.

    If client 10.10.10.1:3000 sends traffic to 203.0.113.1:3000, the source and destination of the traffic are both the same client, so the traffic should be dropped. However, traffic from 10.10.10.1:4000 to 203.0.113.1:3000 is legitimate hairpin traffic and should be allowed.

    Self-IP hairpin filtering drops traffic from 10.10.10.1:3000 or 10.10.10.1:4000 to 203.0.113.1:3000. Self-IP-port hairpin filtering drops traffic from 10.10.10.1:3000 to 203.0.113.1:3000. However, traffic from 10.10.10.1:4000 to 203.0.113.1:3000 is allowed.

NOTE: When enabled, self-IP-port hairpin filtering applies to both TCP and UDP traffic.

- **None**—This option is the default option and filters differently depending on the traffic type:
  - UDP traffic – When this hairpin filtering option is enabled, UDP hairpin traffic is not dropped, even if the UDP traffic addressed to a client's public IP address is from the client's own private IP address. The traffic is allowed even if the source

UDP port is the same as the source UDP port used in the mapping for the client.

- TCP traffic – Self-IP-port hairpin filtering is used for TCP traffic.

NOTE: When enabled, this hairpin filtering option does not filter UDP hairpin traffic. This option still uses self-IP-port filtering for TCP hairpin traffic.

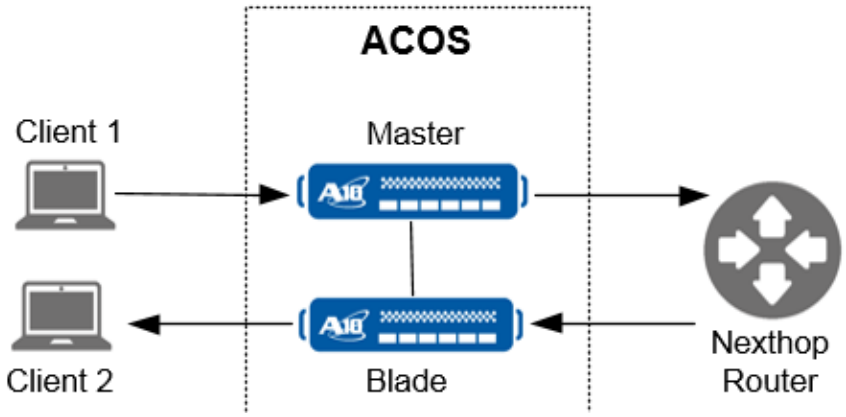
Hairpinning Support for Chassis

This section describes the steps required to configure hairpinning support for A10 Thunder™ Series with dual-processing modules.

Prerequisites:

- Configure a route for the NAT IP address network pointing to the Nexthop Router, and
- Configure the Nexthop Router to loop back traffic destined to the NAT IP address back to the chassis.

Figure 5 : Chassis CGN Hairpinning



In this scenario, two clients reside on two blades. The following summarizes the flow of the traffic:

1. Upon receiving the packet from Client 1, the Master blade performs source NAT-ing and forwards the packet to the Nexthop Router.
2. The Nexthop Router loops back the traffic to the chassis.

3. The Nexthop Router sends the packet back to the chassis which directs it to the other blade.
4. This blade performs destination NAT on the packet.
5. The packet is sent to Client 2.

The following commands show how to configure hairpinning on a CGN chassis:

```
ACOS(config)# cgnv6 nat pool p1 19.9.9.51 19.9.9.52 netmask /24
ACOS(config)# ip route 19.9.9.0 /24 12.10.10.172
```

In this example, 12.10.10.172 is the Nexthop Router address.

Hairpinning Support for CGN Scaleout

ACOS supports hairpin traffic in a multi-node CGN Scaleout. For details on Scaleout CGN hairpin traffic configuration, see the section Configuring Hairpinning in Scaleout CGN in the *Scaleout Configuration Guide*.

User Quotas

User quotas limit the number of NAT port mappings that are allowed for individual internal IP addresses. For example, you can limit each inside IP address to a maximum of 100 TCP NAT ports. Once a client reaches the quota, the client is not allowed to open additional TCP sessions.

When a client has more than one NAT IP assigned, ACOS assigns a user quota for each NAT IP. For example, if a client 192.168.1.1 has two NAT IPs—203.0.113.1 and 203.0.116.1, then each NAT IP can be limited to a maximum of 100 TCP NAT ports.

Before choosing a NAT IP for an internal user, LSN ensures that there are enough ports free on that NAT IP for the user. This guarantees that internal users can use as many ports as possible.

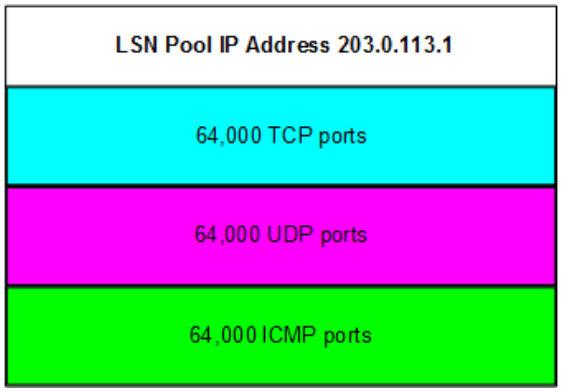
You can configure separate quotas for the following protocols on a global, per-prefix, or individual LSN Limit-ID (LID) basis:

- TCP
- UDP

- ICMP

Each NAT IP has 64,000 TCP ports, 64,000 UDP ports, and 64,000 ICMP ports that can be used for user sessions on the address.

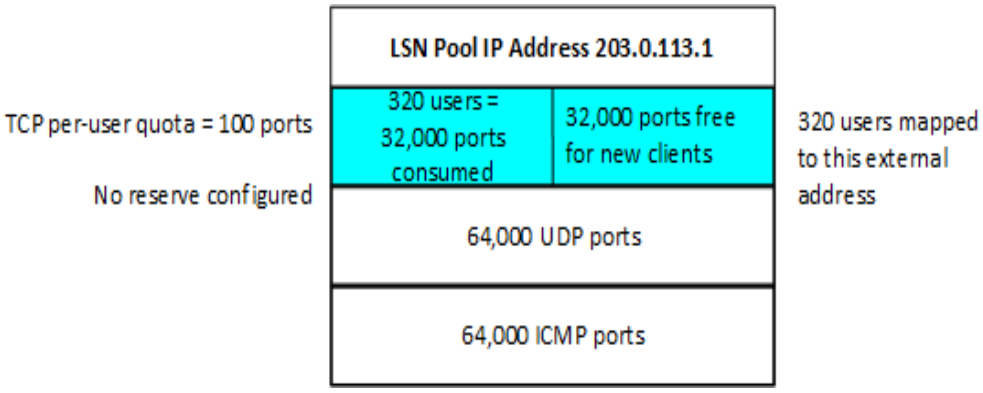
Figure 6 : Protocol Ports Available for Per-User Quota



The per-user quota for a protocol specifies the maximum number of ports a given internal user can use at the same time on a NAT IP. For example, if you set the TCP per -user quota to 100 ports, each internal user can have a maximum of 100 TCP NAT ports allocated on a NAT IP.

In Figure 7, 320 internal users are mapped to a NAT IP. Each of the users consumes 100 TCP ports, leaving 32,000 ports free for new users. In this example, there is room for an additional 320 internal users on the NAT IP.

Figure 7 : Per-User Quota



In the Figure 7, when the user is mapped to the NAT IP, each internal user do not immediately consumes 100 of the NAT IP's TCP ports. If the typical port consumption per user is expected to be lower than the per-user quota, you can also specify a

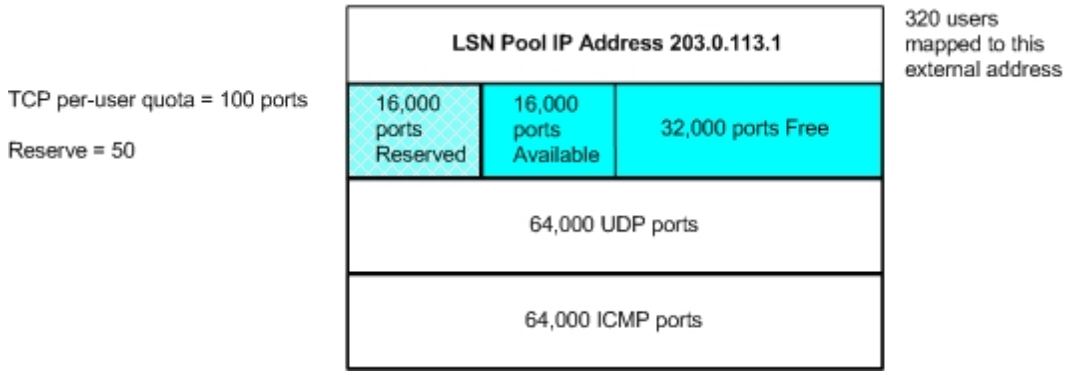
reserve value. When you specify a reserve value, this value allows more internal users to be mapped to the NAT IP.

If the reserve value is not set, the value provided is considered as the reserve value.

NOTE: The reserve value is a subset of the per-user quota.

When you specify a reserve value, each new internal user immediately consumes the number of reserved ports. However, the remaining ports in the user's quota are not consumed unless the user actually needs them. The remaining unconsumed ports are available to new users.

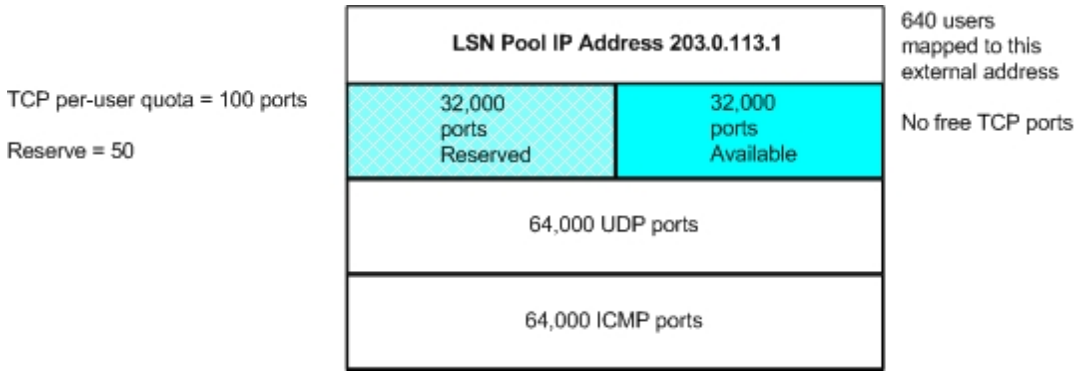
Figure 8 : Per-User Quota with Reserve



In Figure 8, none of the 320 internal users currently mapped to the NAT IP is using more than their reserve value of 50 TCP ports each. This leaves the remaining ports in each user's quota available for new users.

When new users are mapped to the NAT IP, those users receive ports from the free ports. After all free ports are assigned to users, the available ports in the existing users' quotas are assigned to new users. In Figure 9, the external IP address does not have any more free ports. However, none of the users are actually using all of the ports in their 100-port quota. In this example, none of the users are using more than the 50 reserved ports in the quota. Although there are no more free ports, 32,000 ports are still unused and are available for mapping to new internal users.

Figure 9 : Per-User Quota with Reserve - no free ports



If the inside client is not assigned to a NAT IP address, LSN selects an available NAT IP address. This address must meet the following requirements to be used for the client:

- The address must have enough free or available TCP ports to fulfill the configured per-user TCP reserve.
- The configured per-user TCP reserve must not exceed the number of free or available ports on the NAT IP address.
- The address must have enough free or available UDP ports to fulfill the configured per-user UDP reserve.
- The configured per-user UDP reserve must not exceed the number of free or available ports on the NAT IP address.

NOTE: There is a difference between available ports and free ports. You can allocate more than the reserve value, but you cannot allocate more than the user-quota value.

These requirements must be met for TCP, UDP, or ICMP. If the NAT IP address can not meet the requirements, another available address is selected and evaluated for the same requirements. The process continues until there is an available NAT IP address that meets all requirements.

By default, when a client reaches its quota for a protocol, no new translations for that protocol are allowed. To ensure that ports are available for essential services, you can configure an extended quota for the protocol ports that are used by those essential services. For example, to ensure that email service remains available, you

can configure an extended quota for TCP port 25, the standard port used by Simple Mail Transfer Protocol (SMTP).

Extended quotas can be configured on individual LSN LIDs, for individual destination ports. For user quotas in all applications, the regular quota is used first. The extended quota is used only if all regular quota ports are in use, and only for the specified application. The extended quota is always released before the regular quota.

Example:

- TCP user quota = 10
- Extended user quota for TCP port 25 (email) = 5

Table 4 is an example of how ports are used and released with these quotas.

Table 4 : Extended-User-Quota Example

User Connections		**Regular Quota Available**	**Extended Quota Available**
TCP 80 (web)	**TCP 25 (email)**		
No connections	No connections	10	5
No connections	User opens 1 connection	9	5
User opens 4 connections		5	5
	User opens 1 more connection	4	5
User opens 4 more connections (total 8) No additional port 80 are connections allowed.		0	5
	User opens 5 more connections (total 7) No additional port 25 connections allowed.	0	0
User frees 4 connections No additional port 80 are	4 more connections allowed	0	4

Table 4 : Extended-User-Quota Example

User Connections		Regular Quota Available	Extended Quota Available
TCP 80 (web)	TCP 25 (email)		
connections allowed.			
User frees remaining 4 connections		3	5

You can specify between 1 and 65535 internal addresses that can be simultaneously mapped to a public address.

User Quotas for TCP and UDP Sessions

You can configure separate quotas for the following sessions on an individual LSN Limit-ID (LID) basis. These work independently.

- `sessions` – Includes all protocols, such as TCP, UDP, ICMP
- `session-tcp` – For only TCP sessions
- `session-udp` – For only UDP sessions

You can configure the session-tcp and session-udp using the following commands:

```
ACOS(config-lsn-lid)#user-quota session-udp <num>
ACOS(config-lsn-lid)#user-quota session-tcp <num>
```

When you configure all three session quotas, consider the following:

- UDP sessions cannot exceed the `user-quota session-udp` value for UDP traffic.
- TCP sessions cannot exceed the `user-quota session-tcp` value for TCP traffic.
- The total sessions, including TCP, UDP, and other protocols (such as ICMP, and GRE) cannot exceed the `user-quota session` value.

Here are some scenarios with examples of configuring user-quota session, session-tcp, and session-udp.

Example 1

Scenario: The sum of the session-udp quota and session-tcp quota is less than the total session quota to limit UDP and TCP sessions and allow other types of sessions.

```
ACOS(config)#cgnv6 lsn-lid 1
ACOS(config-lsn-lid)#source-nat-pool p1
ACOS(config-lsn-lid)#user-quota session 20
ACOS(config-lsn-lid)#user-quota session-udp 5
ACOS(config-lsn-lid)#user-quota session-tcp 5
```

Here,

- The total session quota is 20.
- The session-udp quota is 5, which means UDP sessions cannot exceed 5.
- The session-tcp quota is 5, which means TCP sessions cannot exceed 5.
- The sum of session-udp (5) and session-tcp (5) is less than the total session quota of 20. This allows the other types of sessions.

Example 2

Scenario: If the session-udp quota + session-tcp quota exceeds the total session quota, TCP and UDP protocol competes for session resources.

```
ACOS(config)#cgnv6 lsn-lid 2
ACOS(config-lsn-lid)#source-nat-pool p2
ACOS(config-lsn-lid)#user-quota session 20
ACOS(config-lsn-lid)#user-quota session-udp 15
ACOS(config-lsn-lid)#user-quota session-tcp 12
```

Here,

- The total session quota is 20.
- UDP sessions can go up to 15, as it does not exceed the total session of 20.
- TCP sessions can go up to 12, as it does not exceed the total session of 20.
- The total of 15 (UDP) + 12 (TCP) exceeds 20, UDP and TCP sessions compete for session resources.

Example 3

Scenario: If the session-udp is greater than the total session quota, the session-udp will not take effect because the UDP session quota cannot exceed the value in the total session quota.

```
ACOS(config)#cgnv6 lsn-lid 2
ACOS(config-lsn-lid)#source-nat-pool p2
```

```
ACOS(config-lsn-lid)#user-quota session 20
ACOS(config-lsn-lid)#user-quota session-udp 25
ACOS(config-lsn-lid)#user-quota session-tcp 12
```

Here,

- The total session quota is 20.
- UDP sessions can go up to 25, exceeding the total session of 20.
- Session-udp (25) will not take effect as it exceeds the total session of 20.

Example 4

Scenario: The user-quota session applies to only the UDP protocol, whereas it is unlimited for TCP and other protocols.

```
ACOS(config)#cgnv6 lsn-lid 1
ACOS(config-lsn-lid)#source-nat-pool p1
ACOS(config-lsn-lid)#user-quota session-udp 5
```

Here,

Maximum number of UDP sessions is limited to 5, whereas for TCP and others, it is unlimited.

Static Port Reservation

A common issue with NAT occurs when an inside user wants to advertise a service on a port. For example, an HTTP server will need to advertise port 80. However, since the NAT IP is shared, only one user per NAT IP can have port 80.

You can allow an inside user to reserve a specific NAT port. In this example, the NAT port 80 would be statically assigned to the user.

NOTE:

- Port reservation is sometimes referred to as port forwarding.
- SIP ALG and Hairpinning do not work with Port reservation.

Exclude Ports from LSN NAT Pools

ACOS lets you exclude a specific port or a range of ports from the LSN NAT pool in both shared and L3v partitions.

Some ports are susceptible to malicious attacks. The remote firewalls or Intrusion Prevention Systems (IPS) with strict security policies on some source ports block such application protocol in order to prevent malicious attacks. When such ports are excluded from LSN NAT pool, they cannot be used by a new session or port reservation.

You can exclude both TCP and UDP ports from LSN NAT pools. The ports can be excluded from non-batch, port batching v1, and port batching v2 LSN NAT pools. The exclude ports must be configured on the same partition in which the NAT pool is configured.

NOTE: The exclude ports do not work on the following:
- One-to-one NAT
- Static NAT
- Fixed NAT

While configuring exclude ports, if the port is being used by a session, the port continues to be in use until the session is closed. After the session is closed, the port is excluded from the NAT pool.

Similarly, if the port is reserved while configuring exclude ports, it continues to be reserved until the port reservation is cleared. After the port reservation is cleared, the port is excluded from the NAT pool.

Configuring Exclude Ports Using CLI

Use the `cgnv6 nat exclude-port` command to exclude ports from the NAT pool.

To exclude a TCP port from the NAT pool, use the following command:

```
ACOS(Config)#cgnv6 nat exclude-port tcp
```

To exclude a UDP port from the NAT pool, use the following command:

```
ACOS(Config)#cgnv6 nat exclude-port udp
```

The following example excludes a specific TCP port from the NAT pool:

```
ACOS(Config)#cgnv6 nat exclude-port tcp
ACOS(config-exclude-tcp-port)#port 1080
```

The following example excludes a range of UDP ports from the NAT pool:

```
ACOS(Config)#cgnv6 nat exclude-port udp
ACOS(config-exclude-tcp-port)#port 1080 to 1090
```

Configuring Exclude Ports Using GUI

To exclude ports using GUI, navigate to **CGN > LSN > Global**.

Perform the following steps:

1. Navigate to **CGN > LSN > Global.**
2. Click **NAT Exclude Port** at the bottom.
3. From the **Protocol** drop-down list, select TCP or UDP.
4. To exclude a specific port from the LSN NAT pool, enter a port number in **Port Start** and leave the **Port End** field blank. Click the save icon.
5. To exclude a range of ports from the LSN NAT pool, enter the starting port range in **Port Start** and the ending port range in **Port End**. Click the save icon.
6. Click +Add to add more ports to exclude from a NAT pool.
7. Click **Update.**

LSN Traffic Inbound Refresh

ACOS provides the option to refresh inbound traffic on LSN, LSN Full-Cone sessions, and Fixed NAT sessions (NAT44 or NAT64 or both NAT44 and NAT64s).

By default the session age is refreshed to prevent session age out, at Outbound (client to server), and Inbound (server to client) traffic.

Inbound refresh is useful for applications with no outgoing UDP traffic. However, allowing inbound refresh allows an external attacker or misbehaving application to keep mapping alive indefinitely. This could be a security risk.

Some data ports are susceptible to malicious attacks. Remote firewalls with strict security policies on the source ports block some applications to prevent malicious attacks. Inbound refresh can be disabled on these ports.

When inbound refresh is disabled for LSN and LSN full cone sessions, the session age refresh is disabled on inbound traffic for the session. Age refresh happens only on outbound traffic.

Configuring Inbound Refresh Using CLI

By default, inbound refresh is enabled.

- To disable NAT inbound refresh, configure the following command at the partition level:

  For LSN sessions:

  ```
  ACOS(config)# cgnv6 lsn inbound-refresh disable
  ```

  For LSN full-cone sessions:

  ```
  ACOS(config)# cgnv6 lsn inbound-refresh-full-cone disable
  ```

- To re-enable inbound refresh, configure the following command:

  For LSN sessions:

  ```
  ACOS(config)# cgnv6 lsn inbound-refresh enable
  ```

  For LSN full-cone sessions:

  ```
  ACOS(config)# cgnv6 lsn inbound-refresh-full-cone enable
  ```

- To view the session statistics for LSN full-cone sessions, use the **`show cgnv6 lsn full-cone-sessions`** command as shown below.

  ```
  ACOS(config)# shows cgnv6 lsn full-cone-sessions
  ```

  Consider the following example outputs:

  Example 1:

  ```
  Total LSN Full-cone Sessions: 1
  ```

```
Prot   Inside Address   NAT Address      Outbnd   Inbnd Pool   CPU Age Flags
-----------------------------------------------
UDP    10.1.1.1:10000  20.1.1.111:10000  1      0     lsn-pool 1   -     -
Total Full-cone Sessions shown: 1
```

In this output, there is one active outbound session. The symbol '-' in the **Age** field indicates that the LSN full-cone session is active i.e., session aging hasn't started yet.

Example 2:

```
Total LSN Full-cone Sessions: 1
Prot   Inside Address   NAT Address     Outbnd   Inbnd   Pool   CPU Age Flags
-----------------------------------------------
UDP    10.1.1.1:10000  20.1.1.111:10000  0      1    lsn-pool 1    x     -
Total Full-cone Sessions shown: 1
```

In this output, there is one active inbound session. The symbol **x** in the **Age** field indicates that the LSN full-cone session is inactive and will not be refreshed even on receiving inbound traffic.

Configuring Inbound Refresh Using GUI

To configure inbound refresh using GUI, navigate to **CGN > LSN > Global**.

Perform the following steps:

1. Navigate to **CGN > LSN > Global.**
2. Select **Disable** for the **Inbound Refresh** field to disable inbound refresh. By default, inbound refresh is enabled.
3. Configure all the other parameters.
4. Click **Update**.

NAT Data Session Aging

The client's data session remains in effect until ACOS detects that the session has ended or until the session ages out due to inactivity.

The following actions are taken on each session type:

- For a TCP session, the data session is removed when ACOS observes the FIN or RST messages exchanged by the two end points of the session. If ACOS does not observe the FIN exchange but the session is idle, the mapping is removed when the session ages out.
- For a UDP session, the data session is removed when the session ages out.
- For an ICMP session, the data session ends when the ICMP reply is received, or when the session ages out.

To configure TCP, UDP, AND ICMP, enter the **`cgnv6 translation`** command and one of the following options:

- **`tcp-timeout`** – Configurable to 60-1500 seconds. The default is 300 seconds.
- **`udp-timeout`** – Configurable to 60-1500 seconds. The default is 300 seconds.
- **`icmp-timeout`** – Configurable to 60-1500 seconds, or **`fast`.** The **`fast`** option uses the SLB
  maximum session life (MSL), which is 2 seconds by default. The default is **`fast`**.

NOTE: Static mappings can be configured, and a static mapping never ages.

NAT Mapping Removal and Full-Cone Behavior

When a NAT data session is removed, removing the NAT mapping that is used by the data session depends on whether full-cone behavior is present. If full-cone behavior is not present, the NAT mapping is removed when the data session is removed. If full-cone behavior is present, the NAT mapping remains in effect until all the data sessions that use the mapping are removed.

For example, if a client uses source port 50000 to connect to two different destinations, the same NAT mapping is used for both data sessions. (This is endpoint-independent mapping.) The NAT mapping is not removed until the data sessions with both destinations have been removed.

LSN maintains the NAT mapping for a full-cone session until the STUN timeout after the final data session ends. By default, the STUN timeout is 2 minutes and is configurable. For information about STUN Timeout, see Changing the STUN Timeout.

By default, full-cone behavior for well-known destination ports (1-1023) is disabled. Full-cone behavior does not apply to ICMP sessions.

One-to-One NAT Based on the Destination IP

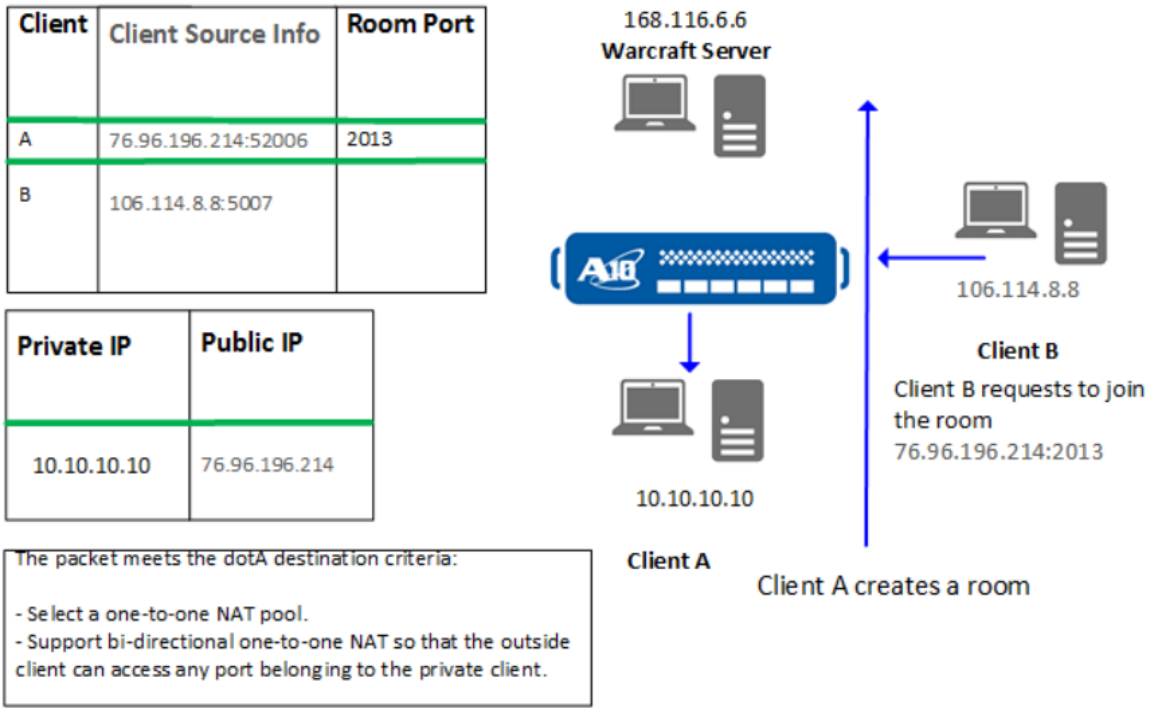
One-to-One NAT supports both NAT44 and NAT64. When a client on a private network connects to applications or services on the Internet, a full, dedicated, public (NAT) IP might need to be reserved for this clients' exclusive use for that service. This public IP will not be shared by any other clients on the private network. The destination IP address of the server that is providing the service or application is used to determine whether the client needs the dedicated, full public IP.

One-to-One NAT allows the ACOS device (responsible for the NAT conversion of the client's private IP to the public IP address) to provide access to the service or application on the Internet using the assigned public IP. The dynamically-assigned, public IP will be exclusively reserved for the client's use.

When the internal client has a unique public IP address, all traffic destined to the same destination IP can reach the client by using any of the client's protocol ports. After all the sessions that use this NAT IP address expire, the NAT IP address is released to the pool based on a configured timeout period.

For example, when Client A with an IP address of 10.10.10.10, and with a NAT address of 76.96.196.214 connects to a Warcraft server, it creates a game room with local port 2013. If one-to-one NAT based on destination criteria is enabled, when the internal client (Client A) connects to a specific server, the client creates a one-to-one NAT address mapping with bi-directional NAT for the inside client. Once this one-to-one mapping is configured, the outside client (Client B) can access any port of the internal client by using the public IP for Client A, as shown in the following graphic.

Figure 10 : Communication with One-to-One NAT



NOTE: For all other services, applications, or destinations, the inside client will continue to use traditional dynamic NAT.

The maximum supported One-to-One NAT IPs vary based on the platform memory.

The following table summarizes the platform memory and the maximum number of One-to-One NAT IPs supported for each platform:

Table 5 : Maximum Number of One-to-One NAT IPs Supported

Platform Memory (GB)	RADIUS Table Size (Entries)
2	20480
4	128K
8	256k
12	512k
16	512k
32	1 Million

Table 5 : Maximum Number of One-to-One NAT IPs Supported

Platform Memory (GB)	RADIUS Table Size (Entries)
64	2 Million
128	2 Million

For information about configuring one-to-one NAT, see Configuring One-to-One NAT.

Limitation

- The maximum number of configurable NAT pools is 8000 NAT pools.
- The maximum number of NAT IP addresses in a single NAT pool is 4096 IPs.
- The logging is supported only for One-to-One NAT44 and NAT64 sessions.

Radius Support

NAT Profile Assignment Based on RADIUS Attribute

You can map an inside client to CGN parameters in an LSN LID that is based on RADIUS. For example, you can use this feature to map clients of a given user type, defined by RADIUS attribute, to a specific CGN pool.

Consider the following issues:

- If the RADIUS attributes that are received for a client do not include an attribute that is assigned to an LSN LID, ACOS can use the default LSN LID to handle the client.
- If port reservation needs to be configured for an inside IP address in the range of an CGN RADIUS profile in a class list, the NAT IP address of the port reservation configuration must be in the range of the NAT pool that is used by the default LSN LID in the CGN RADIUS profile.

ACOS can act as a RADIUS server and as a RADIUS client, depending on the CGN feature you are using and how it is configured. In most cases, ACOS acts as a RADIUS server. Table 6 lists the features for which ACOS acts as a RADIUS server or client.

Table 6 : ACOS RADIUS for CGN

RADIUS Operational Mode	CGN Feature	See...
Server	CGN logging ACOS acts as a RADIUS server to receive client mobile numbers and inserts these numbers in the CGN log messages for those clients.	Traffic Logging Guide for IPv6 Migration
Server	HTTP ALG ACOS acts as a RADIUS server to receive client mobile numbers and inserts these numbers in the HTTP requests from clients. When you configure this feature, you can specify whether to query external RADIUS servers if ACOS does not receive information from those servers. If you enable this option, ACOS acts a client to send the queries.	
Server	NAT profile (LSN LID) selection based on RADIUS ACOS acts as a RADIUS server to obtain client information and selects a NAT profile for the client based on the information. To populate the ACOS RADIUS attribute table for this feature, ACOS must receive RADIUS Accountings-start records that contain the client attribute values.	Continue reading this section.
Client	HTTP ALG If you enable ACOS to query external RADIUS servers if ACOS does not receive	See the description of HTTP ALG in this table.

Table 6 : ACOS RADIUS for CGN

RADIUS Operational Mode	CGN Feature	See...
	information from those servers, ACOS acts a client to send the queries. Note: The client mode of HTTP ALG is not applicable to Fixed-NAT.	

NOTE: RADIUS operation for CGN is not related to RADIUS use as a method to authenticate management access to ACOS. For more information about using RADIUS as an admin AAA method, see *Application Access Management*.

Custom RADIUS Attributes

When you assign a NAT profile to a client based on RADIUS, and if this profile matches the custom attribute that is returned for a client to its name in ACOS, then select the LSN LID mapped to the custom attribute.

For example, ACOS receives a RADIUS Accounting Start record for a client, which contains the following client information:

```
Acct-Status-Type = Start
Acct-Session-Id = 1
Framed-IP-Address = 0x33010133
A10-CGN-Inside-IPv6-Addr = 2001:99:3301:133::1
A10-CGN-Radius-Custom-1 = "cc1"
3GPP-IMSI = 000000001111111111
3GPP-IMEISV = 00000000022222222222
Calling-Station-Id = 12345678901234567890
```

This example uses two RADIUS attributes:

- A10-CGN-Inside-IPv6-Addr – Client IPv6 address
- A10-CGN-Radius-Custom-1 – One of the custom attributes. In the CGN configuration on the ACOS device, the client is assigned to the LSN LID that is mapped to the A10-CGN-Radius-Custom-1 (custom1) attribute. (In the

configuration example later in this section, the custom1 attribute is mapped to LSN LID 1. The client is assigned to that LSN LID 1, and receives the CGN NAT settings configured in it.)

You can assign up to 6 custom attributes to the RADIUS attributes. The RADIUS attribute uses the name that you assign to the custom attribute, not the actual value of that attribute. For example, all clients with A10-CGN-Radius-Custom-1 (custom1) are assigned to the same LSN LID. The value of A10-CGN-Radius-Custom-1 (custom1) is not used.

NOTE: The mapping of custom attributes (such as custom1, custom2, ...custom6) in ACOS is not restricted to the (A10-CGN-Radius-Custom-1, A10-CGN-Radius-Custom-2,... A10-CGNRadius-Custom-6) attributes from RADIUS. You can map any RADIUS attribute to any custom attribute name using the vendor ID and number as part of the mapping.

The length of the custom attributes for RADIUS entries is 64 bytes.

Default LSN LID Selection

You can specify a default LSN LID to use for clients whose RADIUS Accounting Start records do not include one of the custom RADIUS attributes.

Consider the following issues:

- Dynamic Class-List Changes

  Class-list changes do not affect LSN sessions that are already in effect when the class list changes occur.

  For example:

  Some data sessions, user-quota sessions, or full-cone sessions are created for inside user X. Then, the class list is changed in a way that affects X.

  The sessions for X will stay alive as long as there is traffic matching them.

- LSN IP Selection

  The method used for selection of an IP address in an LSN pool does not apply to pool selection in a pool group.

Selection of a pool from in a pool group is always random. After a pool is randomly selected, the configured IP selection method is used to select an IP address from the pool.

For example:

The least-used-strict method is enabled for LSN IP address selection. For a new NAT session:

A pool is randomly chosen from the pool group.

The least-used IP address in that pool is chosen for the new NAT session.

Configuring Platform-based LSN RADIUS Table Size

The LSN RADIUS table's maximum number of entries varies depending on the memory size of the particular ACOS platform. Table 7 lists the memory size and RADIUS table size for each platform.

The LSN RADIUS table size only limits the maximum number of entries supported for each platform. You can choose to configure a custom size for your LSN RADIUS table, not to exceed the maximum.

Table 7 : LSN RADIUS Table - Maximum Number of Entries Per Device

Platform	Memory (GB)	RADIUS Table Size (Entries)
vThunder	4	512000
TH 3030S	16	4000000
TH 3230	14	4000000
TH 4430	32	8000000
AX 3530	64	16000000
TH 5430	64	16000000
TH 5430-11	64	16000000
TH 5435-SPE	64	16000000
AX 5630	128	32000000
TH 6430	128	32000000

Table 7 : LSN RADIUS Table - Maximum Number of Entries Per Device

Platform	Memory (GB)	RADIUS Table Size (Entries)
TH 6435-SPE	128	32000000
TH 6630	128	32000000
TH 14045	256	32000000

To configure the RADIUS table size, use the **`cgnv6 resource-usage radius-table-size`** command. This command configures the total configurable CGNV6 RADIUS table entries.

Configuring RADIUS Accounting-On Requests

Upon receiving a RADIUS accounting-on request, the ACOS device can delete RADIUS table entries associated with the attributes specified in the accounting-on request. After deleting all entries associated with the specified attribute, ACOS will send a RADIUS accounting-response. This allows for users to send accounting-on requests as a status in cases when the user's server may be unable to send accounting-off requests. By default, ACOS will ignore RADIUS accounting-on requests. Use the **`accounting on delete-entries-using-attribute`** option at the RADIUS server configuration level to delete the entries associated with the specified attribute. For details, see the CLI configuration in Configuring RADIUS Accounting Requests.

If ACOS is in the process of deleting RADIUS table entries, then it will ignore any other accounting-on requests related to that attribute that are received. If a RADIUS accounting-start request is received, and its filter attribute data matches the attribute data being deleted, then the accounting-start request will be dropped. The statistics for ignored RADIUS request messages can be viewed using the **`show system radius server statistics`** command. For more information about **`show system radius server statistics`** command, see *Command Line Interface Reference*

Ping Replies from NAT Pool Addresses

By default, ACOS does not reply to ping requests that are sent to NAT addresses (LSN NAT pool addresses). Instead, ACOS drops ping requests that are sent to LSN NAT

pool addresses. You can enable ping replies for LSN NAT pool addresses, which helps you test connectivity to the pool.

You can enable ping replies from NAT addresses on a global basis for IPv4 or IPv6. The setting applies to all IP address pools of the applicable IP version that is configured on the ACOS device.

Consider the following information:

- For security reasons, do the following:
  - Leave the option disabled, except when you need it for testing.
  - Leave the option disabled in environments with live traffic.

This feature applies to LSN NAT pool addresses and to address in NAT pools that are used for standard NAT.

Application Level Gateway

LSN provides Application Level Gateway (ALG) support for the following protocols:

- File Transfer Protocol (FTP)
- Trivial File Transfer Protocol (TFTP)
- Session Initiation Protocol (SIP)
- Real Time Streaming Protocol (RTSP)
- Point-to-Point Tunneling Protocol (PPTP) Generic Routing Encapsulation (GRE)
- IPsec Encapsulating Security Payload (ESP)

By default, ALG support for FTP is enabled, but ALG support for the other protocols is disabled.

NOTE:

- ALG support for protocols other than FTP must be enabled explicitly in the configuration.
- When a full-cone support is enabled for well-known ports, ALG support for TFTP works even if TFTP ALG support is disabled.
- Session synchronization is not supported for ESP.

This section describes the supported application level gateway (ALG) options.

SIP ALG

SIP ALG is disabled by default. You can enable it separately for LSN, NAT64, DS-Lite, and Fixed NAT.

When SIP ALG support is enabled, ACOS creates full-cone sessions to establish NAT mappings for SIP clients, and performs the necessary IP address translations in the SIP packet headers. The full-cone sessions are created for the SIP Contact port and the Real-time Transport Protocol (RTP)/Real-time Control Protocol (RTCP) port.

464XLAT SIP ALG Support

Previously, when a connection is made to an IPv4 resource with the SIP protocol, address:port information is transferred within the payload. The client-XLAT performs partial stateless translation since it uses different IPv6 prefixes for CLAT-side and PLAT-side IPv4 addresses.

As an IPv6 packet comes to the provider, it's consumed by the NAT64 service. Since NAT64 ALG checks the packet and its payload, instead of identifying the expected IPv6 address, the private IPv4 address in the protocol payload is recognized.

This causes the ALG process to fail. Though the transport header is translated by NAT64, the sent packet becomes unusable since the payload contains wrong information.

By leveraging 464XLAT SIP ALG support, it combines stateless IPv4-IPv6 translation on the end device (CPE) with stateful IPv6-IPv4 translation on the provider side (usual NAT64), allowing limited IPv4 access services being deployed to IPv6-only edge networks without the need of encapsulation.

This is achieved by NAT64 SIP ALG support in modifying the SIP payload in 464XLAT. On the outbound traffic, the private IPv4 address in the payload is translated to a NAT IP address. On the inbound traffic, the NAT IP address in the payload is translated to the private IPv4 address instead of IPv6 address.

IPsec ESP

LSN supports passthrough of Encapsulating Security Payload (ESP) packets. ESP is a protocol that is used by IP security (IPsec) to secure IP packets. When you use this protocol, the source IP address is changed and the NATted IP address becomes the same as that of the IPSec control session.

LSN requires persistence of internal-IP to NAT-IP mappings. IP mappings are determined prior to ESP sessions, so it is not guaranteed that each internal IP address will be mapped to different NAT IP address. Therefore, it is possible for an IP conflict to occur if multiple internal clients are connecting to the same external VPN server. If there is an IP conflict, the second connection is dropped.

Source NAT for ICMP Error Messages

By default, ACOS does not translate the source IP addresses of ICMP error messages that are sent by inside routers into NAT addresses. As a result, if the messages reach the outside server, the outside server may be unable to determine the source of the messages.

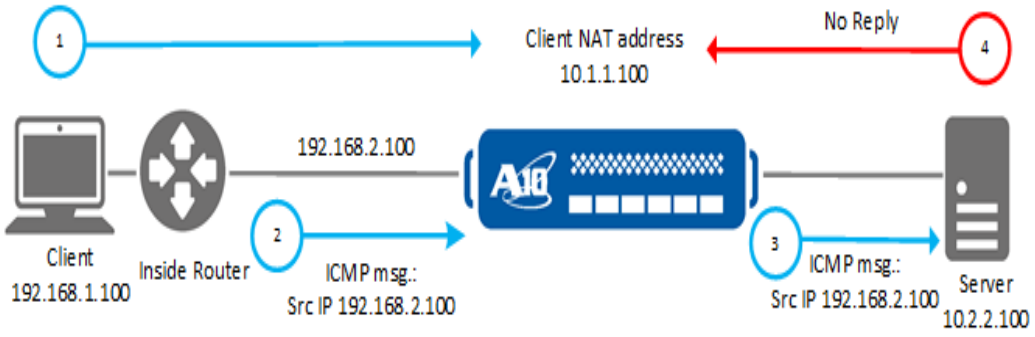
When you can enable source NAT for ICMP messages from inside routers, ACOS uses the inside client's NAT address as the source address for ICMP messages that are related to the client's session. Figure 11 and Figure 12 illustrate the behavior of the ACOS device when this option is disabled or enabled.

NOTE: You can use source NAT for ICMP Error Messages only for IPv4 NAT client and NAT addresses (NAT444/LSN).

Source NAT for ICMP Messages Disabled (default)

Figure 11 illustrates the default behavior when a router in the inside network sends an ICMP error message to a server on the outside network.

Figure 11 : Source NAT for ICMP Messages Disabled (default)



This is the traffic flow for this example:

1. The inside client (192.168.1.100) sends a request to the outside server.
2. ACOS maps the client to NAT address 10.1.1.100.

   From the perspective of the outside server, 10.2.2.100, the client has an IP address of 10.1.1.100.
3. During the session, an inside router, through which the client is communicating with the outside server, sends an ICMP error message to the server.

   The source IP address of the ICMP message, 192.168.2.100, is in the inside network.
4. On the ACOS device, the source NAT for ICMP error messages is disabled.
5. The ACOS device sends the inside router's ICMP message to the outside server, without translating the source IP address into a NAT address.

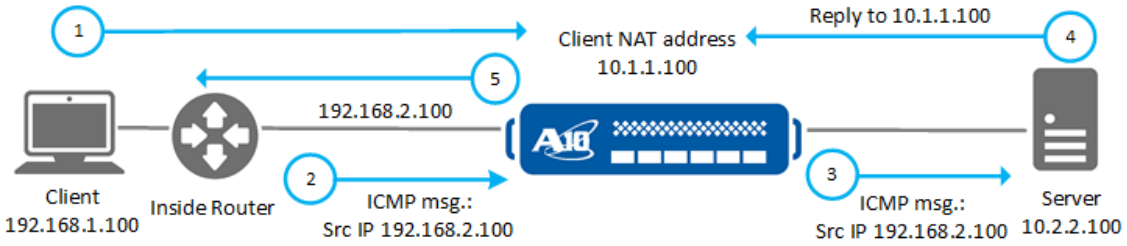
   From the perspective of the outside server, 10.2.2.100, the ICMP message comes from an IP address that might not be routable or might overlap with a local address.
6. The server does not have a session with client 192.168.2.100.

7. The server drops the ICMP message, and no reply is received by the inside router or client.

Source NAT for ICMP Messages Enabled

Figure 12 illustrates the behavior when you enable source NAT for ICMP error messages.

Figure 12 : Source NAT for ICMP Messages Enabled



This is the traffic flow for this example:

1. The inside client, 192.168.1.100, sends a request to the outside server.
2. ACOS maps client to NAT address 10.1.1.100.

   From the perspective of the outside server, 10.2.2.100, the client has IP address 10.1.1.100.

3. During the session, an inside router, through which the client is communicating with the outside server, sends an ICMP error message to the server.

   The source IP address of the ICMP message, 192.168.2.100, is in the inside network.

4. ACOS maps the inside router's IP address to the client's NAT address.
5. ACOS forwards the ICMP message by using the client's NAT address as the source IP address.

   From the outside server's perspective, the ICMP message appears to come from client 10.1.1.100.

6. If applicable, the server sends a reply to the ICMP message to the client's NAT address.

7. ACOS translates destination IP address of the reply back into the inside router's IP address and forwards the reply to the router.

LSN Support for SCTP

When Static NAT is configured, there is support for Stream Control Transmission Protocol traffic. SCTP allows for multi-homing and multiple streams in a single connection. For each SCTP connection, there is a maximum of 8 network addresses for each the client and the server. SCTP connections can be established when Static NAT is configured for all the associated IP addresses, although SCTP timeouts are configured separately. For more information about Static NAT, see the System Configuration and Administration Guide.

Either the inside or the outside is able to initialize an SCTP connection, also known as an SCTP association, via a 4-way handshake. First an INIT packet is sent, creating a half-open session. The INIT packet can contain multiple IPv4 addresses to support multi-homing. A response is received in the form of an INIT-ACK packet that contains a State Cookie parameter. After receiving the INIT-ACK packet, the initiator sends a COOKIE-ECHO packet. A response of a COOKIE-ACK packet fully establishes the SCTP connection. This 4-way handshake may contain embedded information about IP or port details in order to create another SCTP connection for incoming traffic.

SCTP packets consist of a common header and control/data chunks which support multiple streams in a single connection. Following the chunks are optional parameters which are used to specify additional IP addresses for multi-homing. These IPv4 addresses can be included in the INIT or INIT-ACK packets. They are NATed based on the Static NAT configuration, and then they are added to the session.

Once an SCTP connection is established, either the inside or the outside can send additional chunks to modify the connection. An ASCONF chunk is set to either add an IPv4 address to the connection, remove an IPv4 address from the connection, or change the primary address. If payload protocol restriction is enabled, DATA chunks are forwarded only if they belong to the permitted protocol. To avoid a flood of SCTP packets during an open session, packet rate limiting can be configured. Abort and Shutdown chunks terminate the SCTP session.

NOTE:

- Currently SCTP over LSN does not support IPv6 addresses for multi-homing.
- Hairpinning for SCTP is not supported.
- Payload protocol restriction and packet rate-limiting is only supported on CFW platforms.

To enable support for SCTP, configure an IP NAT inside and an IP NAT outside on the desired interfaces. Once the IP NAT inside and outside are configured, configure Static NAT.

CGN Deployments with L3V Inter-partition Routing

ACOS supports L3V in CGN deployments.

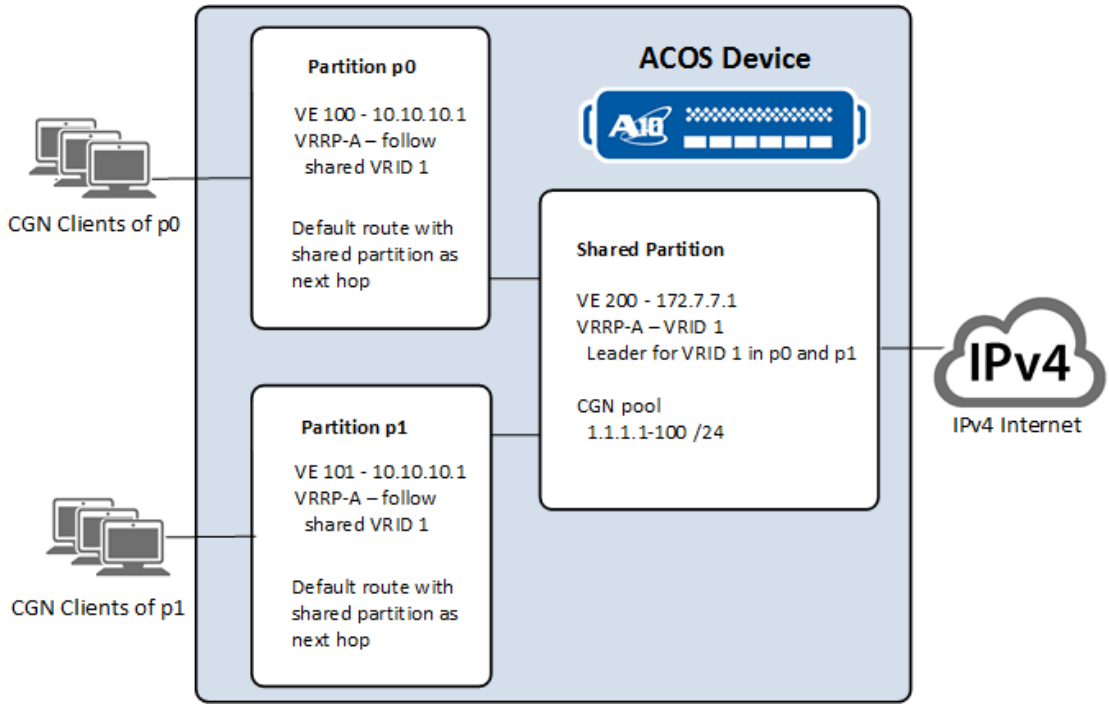
Remember the following requirements when you use CGN, DS-Lite, or NAT64 with L3V inter-partition routing:

- All the inside (private) users must be on private, not shared, partitions.
- You can use only shared NAT pools.

Shared NAT are marked as shared resources and can be shared with all partitions, a single partition, or a partition group. This allows you to specify which private partitions can use which NAT pools.

Figure 13 shows an example L3V deployment for CGN.

Figure 13 : CGN with Inter-Partition Routing



This example has two private partitions with L3V enabled, each with its own CGN clients. Each private partition is configured with the same IP address space.

Client traffic is received by the private partitions on their VLAN Virtual Ethernet (VE) interfaces. A CGN pool in the shared partition is used by each of the private partitions for client NAT mappings. Each private partition has a default route whose next hop is the shared partition.

VRRP-A is used for redundancy, but the second ACOS device is not shown. The CGN pool addresses are backed up by VRRP-A, and each private partition is configured to base its VRRP-A Active/Standby state on the state of the shared partition's VRID.

Destination NAT Port Translation

To enhance granular control of ports, destination port can be translated using the **`cgnv6 port-list`** command.

The port-list contains the mapping between original ports and translated ports. To enable port translation, bind the port-list to an lsn-rule-list, which is then bound to an lsn-lid.

NOTE: Once the port-list is bound to an lsn-rule-list, no changes can be made to the port-list. To make changes to a port-list, unbind it from the lsn-rule-list first.

Destination NAT occurs when the following two conditions are met:

- There is a port-list bound to the lsn-rule-list action,
- In the port-list, there is a translated port associated to the destination port.

To configure a port-list, use the following commands:

```
ACOS(config)# cgnv6 port-list abc
ACOS(config-port-list)# original-port 80 to translated-port 8080
ACOS(config-port-list)# original-port 5353 to translated-port 53
```

To bind the port-list to an lsn-rule-list, use the following commands:

```
ACOS(config)# cgnv6 lsn-rule-list r1
ACOS(config-lsn-rule-list)# default
ACOS(config-lsn-rule-list-default)# tcp port 1 to 65535 action dnat ipv4-list i port-list abc
ACOS(config-lsn-rule-list-default)# udp port 1 to 65535 action dnat ipv4-list i port-list abc
```

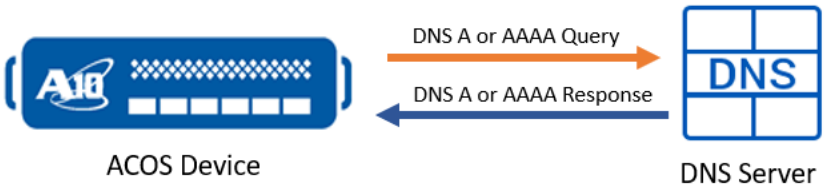
NOTE: Port-list can be configured only if there is an ipv4-list configured for dnat.

To view the information of configured port-lists, use the **`show running-config`** command.

CGNv6 Domain-list DNS Resolution

When a new TCP connection request is received from a NAT client that matches the LSN-Rule-List, the following CGNv6 DNS resolution process is initiated:

1. ACOS sends a DNS query request (both A and AAAA records) to the DNS Server for resolution.



2. ACOS waits for a response from the DNS Server.
3. If the query fails, ACOS initiates a retry within 3 seconds. It immediately resends 5 consecutive queries (for both A and AAAA records) at 3-second intervals.
4. If the query fails even after retrying 5 times, ACOS stops retrying and waits for the query fail-interval (30 seconds by default) before initiating the next DNS query process.

   The query fail-interval is the time in seconds that ACOS waits to restart the next DNS query process for a domain since the last failure. Steps 1, 2, and 3 constitute a DNS query process.

5. The above-mentioned process repeats until a successful DNS response is received.

   In this resolution process, the DNS query interval (10 minutes by default) is the time between DNS queries if the previous query is successful.

NOTE:

- One retry cycle has a maximum of 5 record-A queries and 5 record-AAAA queries.
- If both records A and AAAA fail, or if only record A is successful, ACOS waits for the fail-interval before the next retry cycle.
- If both records A and AAAA are successful, or if only record AAAA is successful, ACOS waits for the query interval before the next query.

Configuring CGNv6 Domain-Lists

The **domain-list** and **cgnv6 global domain-list** commands allow you to configure parameters (query interval and fail-interval) for the CGNv6 domain lists at the domain level and global system level respectively. Additionally, using the **cgnv6 global domain-list** command, you can also disable AAAA query records in the CGNv6 domain-list query, so that only A query records are sent for resolution.

As mentioned previously, the default query interval is 10 minutes. This prolonged query interval may sometimes result in a disruption of service. Setting appropriate parameter values (for query interval and fail-interval) can improve the DNS resolution time and mitigate service outages as well.

CLI Configuration

- To configure the query interval and fail-interval at the domain level:

```
ACOS(config)# domain-list <list-name>
ACOS(config-domain-list:list-name)# <domain-name> interval <minutes>
fail-interval <seconds>
```

The following example command sets the query interval to 4 minutes and the fail-interval to 10 seconds for the domain `test.a10.com`.

```
ACOS(config)# domain-list mylist
ACOS(config-domain-list:mylist)# test.a10.com interval 4 fail-interval
10
```

- To configure the query interval at the global level:

```
ACOS(config)# cgnv6 global domain-list interval <minutes>
```

The following example command sets the global query interval to 2 minutes:

```
ACOS(config)# cgnv6 global domain-list interval 2
```

- To configure the fail-interval at the global level:

```
ACOS(config)# cgnv6 global domain-list fail-interval <seconds>
```

The following example command sets the global fail-interval to 20 seconds:

```
ACOS(config)# cgnv6 global domain-list fail-interval 2
```

- To disable the AAAA query record for each domain in the CGNv6 domain list:

```
ACOS(config)# cgnv6 global domain-list aaaa-query disable
```

Key Considerations:

- Global configurations set using the **`cgnv6 global domain-list`** command are only effective for domains configured with default parameters. This implies that if you explicitly set the query interval and failure interval at the domain level (using the **`domain-list`** command), the global configuration will not take effect.
- Ensure that the query interval and fail-interval are configured only once for a specific domain in one domain list. Configuring these parameters in multiple domain lists may result in undefined behavior.

Configuring Large Scale Network Address Translation

To configure CGN by using the GUI:

1. Configure NAT pools and, optionally, pool groups. Navigate to:
   - **CGN > LSN > LSN Pools**
   - **CGN > LSN > Pool Groups**
2. To configure LSN Limit IDs (LIDs), navigate to **CGN > LSN > LSN-LID**.

   For each LID, specify the NAT pool to use. You can also set user quotas for the LID.
3. To import or configure class lists for the user subnets that require LSN, navigate to **CGN > LSN > Class Lists**.

   A class list is a list of internal subnets or hosts. In a class list, you can bind each internal subnet to an individual LSN LID.
4. Navigate to **CGN > LSN > Interfaces** to complete one of the following tasks:
   - Enable the inside NAT on the interface that is connected to the internal clients.
   - Enable the outside NAT on the interface that is connected to the Internet**.**
5. To bind a class list to the configuration for use with LSN, navigate to **CGN > LSN >**

Global.

The class list will apply to packets from the inside NAT interface to the outside NAT interface. Only a single class list can be used for this purpose.

The following list provides information about some additional options for LSN:

- To configure traffic logging, see the Traffic Logging Guide for IPv6 Migration.
- To configure matching and traffic handling based on destination, see Destination Based NAT.
- To configure Port Control Protocol (PCP), see Port Control Protocol for LSN.
- To configure static mappings, use the following page: **CGN > Static Mapping**
- For other optional settings, use the following page: **CGN > LSN > Global**
- For other LSN options, see the CLI instructions later in this chapter.

To configure LSN using CLI, see the following sections.

Configure LSN NAT pools

1. To configure an LSN NAT pool, enter the following commands

```
ACOS(config)#cgnv6 nat pool cgnpool1 192.168.1.1 192.168.1.10 netmask
/24
```

192.168.1.1 is the beginning public IP address and 192.168.1.10 is the ending public IP addresses in a range to be mapped to internal addresses. The **netmask** option specifies the subnet mask or mask length for the addresses. For additional options, see the *Command Line Reference* guide.

2. To configure NAT Pool exhaustion logging severity, enter the following commands:

```
ACOS(config)# cgnv6 logging nat-quota-exceeded level warning
or
ACOS(config)# cgnv6 logging nat-resource-exhausted level warning
```

When a NAT pool is exhausted, either because the quota is exceeded or else there are no resources left, an error message is logged. The error message's logging

level can be configured so that NAT pool exhaustion is flagged as “Critical,” “Warning,” or “Notice.”

3. To clear the sessions that use a pool, follow these steps:
   a. If you need to modify a pool used for LSN, all sessions using that pool must be cleared first.
   b. To remove the sessions that use a pool, remove the pool from any pool groups and LIDs that use the pool.
   c. Enter the following command to clear the sessions:

```
ACOS(config)# clear cgnv6 lsn all-sessions poolp1
```

   d. Reconfigure the pool.
   e. Add the pool to the pool groups and LIDs that use the pool.

NOTE: You can remove a NAT pool that is associated with a stuck session only by forcing the clear operation. This command clears this session without having to reload your ACOS device.

You must wait until the NAT pool sessions clearing command completes before configuring another command.

Configure LSN Limit IDs (LIDs)

For each LID, specify the NAT pool to use. You can also set user quotas for the LID.

1. Enter the following commands at the global configuration level of the CLI:

```
ACOS(config)# cgnv6 lsn-lid 22
```

2. Enter the following command to binds an LSN NAT pool to the LID:

```
ACOS(config-lsn lid)# source-nat-pool LSN_Pool1
```

3. Enter the following command to configures the per-user mapping quota for each type of protocol supported for LSN (TCP, UDP, or ICMP):

```
ACOS(config-lsn lid)# user-quota tcp 100
ACOS(config-lsn lid)# user-quota udp 300 reserve 100
ACOS(config-lsn lid)# user-quota icmp 10
```

```
ACOS(config-lsn lid)# extended-user-quota tcp service-port 25 sessions
3
```

The **user-quota** option specifies the maximum number of sessions allowed per client. There is no default user quota.

For **tcp** or **udp**, the **reserve** option allows you to specify how many ports to reserve on a NAT IP for each user. If unspecified, the reserve value is the same as the user-quota value.

The **service-port** option specifies the Layer 4 protocol port of the service.

The **sessions** option specifies how many extended sessions are allowed for the protocol port. There is no default extended user quota.

4. To apply the user quotas to all clients in the specified network prefix, enter the following command:

```
ACOS(config-lsn lid)# user-quota-prefix-length 22
```

For more information, see User Quotas Based on IPv6 Prefix.

5. Enter the following command to override NAT for traffic that matches the class list:

```
ACOS(config-lsn-lid)# override drop
```

Instead of performing NAT for matching traffic, LSN performs one of the following actions:

- **drop** – Drops the traffic.
- **none** – Apply source NAT if configured. (This is the default.)
- **pass-through** – Forwards the traffic without performing NAT.

Configure Class Lists for User Subnets that Require LSN

A class list is a list of internal subnets or hosts. In a class list, you can bind each internal subnet to an individual LSN LID.

1. Enter the following command at the global configuration level of the CLI:

```
ACOS(config)# class-list list1
```

This command changes the CLI to the configuration level for the class list. The **list-name** option adds the list to the running-config. If the list is large, you use the **file** option to save the list to a file. In this case, the list entries are not displayed in the running-config.

2. Enter the following class-list-related command:

```
ACOS(config-class list)# 5.5.5.0/24 lsn-lid 5
```

5.5.5.0 /24 specifies the internal host or subnet address and network mask length. **lsn-lid 5** is the LID number.

Bind a class-list to the LSN Feature

The class list applies to packets from the inside NAT interface to the outside NAT interface. There can be a maximum of one class list for this purpose.

Enter the following command to bind the class list to the LSN feature:

```
ACOS(config)# cgnv6 lsn inside source class-list list1
```

Enable Inside NAT on the Interface Connected to the Internal Clients

Enter one of the following commands to enable inside NAT on the interface that is connected to internal clients:

```
ACOS(config)# interface ethernet 4
ACOS(config-if:ethernet:4)# ip nat inside
ACOS(config-if:ethernet:4)# exit
```

This command is entered at the configuration level for the interface.

Enable Outside NAT on the Interface Connected to the Internet

Enter one of the following commands to enable outside NAT on the interface that is connected to the Internet:

```
ACOS(config)# interface ethernet 4
ACOS(config-if:ethernet:4)# ip nat outside
```

```
ACOS(config-if:ethernet:4)# exit
```

This command is entered at the configuration level for the interface.

NAT Pool Utilization

The NAT Pool Utilization statistics provide detailed NAT pool utilization metrics for dynamic NAT. It provides real-time and accurate visibility to manage the port usage and remaining capacity at a pool level. It helps to efficiently manage subscribers across different devices, particularly in Scaleout deployments where IPs are distributed across multiple nodes.

These statistics helps making decisions regarding scaling, capacity planning, and monitoring.

The `show cgnv6 nat pool utilization` command displays the TCP and UDP port usage percentage and user utilization at the NAT pool level. This information serves as a key indicator of resource availability within the NAT pool.

The following SNMP OID can be used to view the NAT pool utilization percentage. For more information, see *MIB Reference*.

```
1.3.6.1.4.1.22610.2.4.8.717.1.1.1
```

To view the NAT pool utilization statistics, use the following command:

```
ACOS(config)# show cgnv6 nat pool utilization
```

Additional Configuration Options

The following sections describe additional configuration options.

The following topics are covered:

Configuring Static Mappings

Enter the following command to optionally configure static mappings for a range of protocol ports for an internal address:

```
ACOS(config)# cgnv6 lsn port-reservation inside 1.1.1.1 22 33 nat 11.1.1.1
22 33
```

This command is entered at the global configuration level of the CLI.

You can specify the internal IP address, the range of internal protocol port numbers, the public IP address to map to the internal IP address, and the range of public protocol port numbers to map to the range of internal protocol port numbers.

Configuring Endpoint-Independent Filtering (EIF) and Mapping (EIM)

By default, full-cone support (EIM Mapping and EIF) is disabled. You can enable EIM and EIF individually for any port range.

Consider the following information:

- Disabling full-cone NAT for all destination ports, including the well-known ones, is equivalent to enabling Symmetric NAT.
- For a destination port or range, do not combine an enabled EIF with a disabled EIM, because EIM filtering always fails when packets are dropped.
- The following combinations of EIM and EIF are not supported for the same destination port or port range:
  - For a given destination port or range, you cannot enable EIF where EIM is disabled.

    For example, you would enter the following command:

    ```
    ACOS(config)# cgnv6 lsn endpoint-independent-filtering tcp
    ```

    The following commands illustrate that EIF for TCP ports 2000 to 3000 is now enabled:

    ```
    ACOS(config)# cgnv6 lsn endpoint-independent-filtering tcp
    ACOS(config-eif-tcp)# port 2000 to 3000
    ```

  - You cannot enable EIF for a port where EIM is disabled.

NOTE: When configuring IP stateful firewall and enabling EIF on Layer 4 ports, some of the services behind the firewall are accessible to outside clients. In this situation, the inside services on the EIF enabled L4 ports become the destination ports as the traffic originates from the outside clients. With IP stateful firewall configured then, EIF configuration should only be done for the inside (destination) Layer 4 ports.

Configuring Endpoint-Independent Filtering

The commands in the following configuration file excerpt configure EIF:

```
ACOS(config)# cgnv6 lsn endpoint-independent-filtering tcp
ACOS(config-eif-tcp)# port 2000 to 3000
ACOS(config)# exit

ACOS(config)# cgnv6 lsn endpoint-independent-filtering udp
ACOS(config-eif-udp)# port 2000 to 3000
ACOS(config)# exit
```

Configuring Endpoint-Independent Mapping

NOTE: By default EIM and EIF are disabled for all ports.

The commands in the following configuration file excerpt configure EIM:

```
ACOS(config)# cgnv6 lsn endpoint-independent-mapping tcp
ACOS(config-eim-tcp)# port 2000 to 3000
ACOS(config)# exit

ACOS(config)# cgnv6 lsn endpoint-independent-mapping udp
ACOS(config-eim-udp)# port 2000 to 3000
ACOS(config)# exit
```

Configuring Full-cone NAT Support

You cannot directly configure full-cone support. To configure full-cone, you must configure EIM and EIF separately.

Changing the STUN Timeout

After the session sends, LSN maintains the NAT mapping for a full-cone session for the duration of the STUN timeout. If the client requests a new session for the same port before the mapping times out, the mapping is used again for the new session. If the mapping is not used again before the STUN timeout expires, the mapping is removed.

NOTE: The default STUN timeout is 2 minutes.

Enter the following command to change the STUN timeout:

```
ACOS(config)# cgnv6 lsn stun-timeout tcp port 22 to 33 50
```

For more information about configuring the STUN timeout for SIP, see SIP ALG.

Configuring the IP Selection Method

The method used by LSN to select an IP address from an LSN NAT pool is configurable on a global basis.

Enter the following command to specify the method for LSN IP address selection in a pool:

```
ACOS(config)# cgnv6 lsn ip-selection round-robin
```

You can replace the **method** option with any of the options in the list in Configuring the IP Selection Method. The method you specify applies to all LSN pools.

NOTE: The least-used-strict option is not applicable to ICMP.

The IP address selection method applies only to the IP addresses in individual pools. The method does not apply to selection of pools in a pool group. LSN randomly selects a pool from in a pool group, then uses the configured IP address selection method to select an address from in the pool.

Configuring One-to-One NAT

The following procedure illustrates a sample configuration.

1. Define a NAT pool called pool_ 1 and specify the IP address of the server to be used by one-to-one NAT:

```
ACOS(config)# cgnv6 one-to-one pool pool_1 11.1.1.1 netmask /24
```

   You must specify the one-to-one keyword if you plan to use the pool for one-to-one NAT. Once specified, the one-to-one NAT pool can only be used for one-to-one NAT purposes.

2. Enter the following commands to configure a NAT pool group for one-to-one NAT:

```
ACOS(config)# cgnv6 one-to-one pool pool_1 11.1.1.1 netmask /24
ACOS(config)# cgnv6 one-to-one pool pool_2 11.1.2.1 netmask /24
ACOS(config)# cgnv6 one-to-one pool pool_3 11.1.3.1 netmask /24
ACOS(config)# cgnv6 one-to-one pool-group group_1to1
ACOS(config-pool-group:group_1to1)# member pool_1
ACOS(config-pool-group:group_1to1)# member pool_2
ACOS(config-pool-group:group_1to1)# member pool_3
```

3. In the LSN rule-list called `myrules`, define the destination criteria for one-to-one NAT:

```
ACOS(config)# cgnv6 lsn-rule-list myrules
ACOS(config-lsn-rule-list)# ip 61.1.1.100/32
ACOS(config-lsn-rule-list-ip)# tcp port 0 action one-to-one-snat pool
pool_1
```

   In the above configuration, when TCP traffic is sent to server 61.1.1.100, it will trigger the one-to-one NAT process by using the one-to-one NAT pool called pool_1.

   a. You can specify a NAT pool-group called group_1to1 to use for one-to-one NAT:

```
ACOS(config-lsn-rule-list-ip)# tcp port 0 action one-to-one-
snat pool group_1to1
```

```
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# exit
```

   b. You must bind your LSN rule-list called `myrules` to the LSN LID:

```
ACOS(config)# cgnv6 lsn-lid 1
```

```
ACOS(config-lsn-lid)# lsn-rule-list destination myrules
ACOS(config-lsn-lid)# exit
```

4. To configure a timeout value of 10 minutes for your one-to-one NAT mapping, enter the following command. When no active one-to-one NAT sessions exist, the mapping will expire at the end of the time period:

```
ACOS(config)# cgnv6 one-to-one mapping-timeout 10
```

5. To display your configured mappings of one-to-one NAT, enter the following show command (with or without filters):

```
ACOS(config)#show cgnv6 one-to-one mappings
>Inside IPv4 Address    Inside IPv6 Address     NAT Address     Sessions
Age    Pool
-------------------------------------------------------------------------
----------------
>-                      3ff7::85                11.1.1.130      0
600    pool_1
>10.1.1.2               -                       11.1.1.129      0
600    pool_1
Total One-to-One NAT Mappings: 2
```

NOTE: A hyphen (‘-’) in the Age column means that the associated mapping is currently being used by some data sessions and will not expire. When all relevant data sessions complete, the hyphen will be replaced by a timeout value in seconds and will expire in accordance to the configured timeout value.

The options related to ADP partitions are valid only if the command is entered in the shared partition. The options are not displayed in private partitions.

6. To display your one-to-one NAT pool statistics of total, used, and free address numbers for a one-to-one NAT pool, enter the following command:

```
ACOS# show cgnv6 one-to-one pool statistics
Pool    Total Address   Used Address            Free Address
-----------------------------------------------------------
pool_1  255             155                     100
pool_2  255             30                      225
```

7. To display one-to-one NAT statistics related to your allocated, freed, or failed mappings, enter the following command:

```
ACOS# show cgnv6 one-to-one statistics
Total One-to-One Mapping Allocated:     23456
Total One-to-One Mapping Freed: 23000
One-to-One Mapping Allocation Failure:  100
…
```

8. To enable logging for one-to-one NAT, enter the following commands:

```
ACOS(config)#cgnv6 template logging log_template_name
ACOS(config-logging:log_template_nam)#log one-to-one-nat sessions
```

9. To clear one-to-one NAT statistics, enter the following command:

```
ACOS# clear cgnv6 one-to-one statistics
```

10. To clear one-to-one NAT mappings, enter the following command:

```
ACOS# clear cgnv6 one-to-one mappings
```

NOTE: You can optionally filter this output by specifying an inside address, NAT address, or a particular pool.

When using the **clear** command, you might see the following behavior:

- One-to-one mapping is cleared only in the current network partition.
- Data sessions are cleared first before the mapping is cleared.
- If you one-to-one NAT mappings are not cleared with the first attempt, wait until the data sessions are cleared, before attempting the clear command a second time.
- If live traffic is being processed, your one-to-one NAT mapping may not be cleared.

Assigning CGN Parameters to Clients Based on RADIUS Attributes

1. Configure the resources that you want to assign to clients based on RADIUS by configuring CGN pools to which the inside clients will be assigned:

```
ACOS(config)# cgnv6 nat pool cgnpool1 192.168.1.1 192.168.1.10 netmask
/24
```

```
ACOS(config)# cgnv6 nat pool cgnpool2 192.168.2.1 192.168.2.10 netmask
/24
ACOS(config)# cgnv6 nat pool cgnpool3 192.168.3.1 192.168.3.10 netmask
/24
ACOS(config)# cgnv6 nat pool cgnpool4 192.168.4.1 192.168.4.10 netmask
/24
ACOS(config)# cgnv6 nat pool cgnpool5 192.168.5.1 192.168.5.10 netmask
/24
ACOS(config)# cgnv6 nat pool cgnpool6 192.168.6.1 192.168.6.10 netmask
/24
ACOS(config)# cgnv6 nat pool cgnpool4 192.168.7.1 192.168.7.10 netmask
/24
```

2. Configure a separate LSN LID for each set of resources (for example, for each pool).

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn lid)# source-nat-pool cgnpool1
ACOS(config-lsn lid)# exit
ACOS(config)# cgnv6 lsn-lid 2
ACOS(config-lsn lid)# source-nat-pool cgnpool2
ACOS(config-lsn lid)# exit
ACOS(config)# cgnv6 lsn-lid 3
ACOS(config-lsn lid)# source-nat-pool cgnpool3
ACOS(config-lsn lid)# exit
ACOS(config)# cgnv6 lsn-lid 4
ACOS(config-lsn lid)# source-nat-pool cgnpool4
ACOS(config-lsn lid)# exit
ACOS(config)# cgnv6 lsn-lid 5
ACOS(config-lsn lid)# source-nat-pool cgnpool5
ACOS(config-lsn lid)# exit
ACOS(config)# cgnv6 lsn-lid 6
ACOS(config-lsn lid)# source-nat-pool cgnpool6
ACOS(config-lsn lid)# exit
ACOS(config)# cgnv6 lsn-lid 7
ACOS(config-lsn lid)# source-nat-pool cgnpool7
ACOS(config-lsn lid)# exit
```

NOTE: Each LID is mapped to a different CGN pool.

3. Add the resources to the LSN LIDs.

   Each LID should contain the unique set of resources to allocate to a given set of clients.

4. Configure an IP list that specifies the RADIUS server addresses or subnets from which to receive the mobile number information.

   The following commands configure the IP list that specifies the external RADIUS server addresses:

```
ACOS(config)# ip-list RADIUS_IP_LIST
ACOS(config-ip list)# 9.9.9.9 to 9.9.9.10
ACOS(config-ip list)# exit
```

   These servers send the AAA information to the ACOS device.

5. Configure the ACOS device to act as a RADIUS server, so that it can receive RADIUS Accounting requests that include the client RADIUS attributes.

NOTE: The cgnv6 lsn radius server CLI command is deprecated. If this deprecated command is used in old configurations, it must be replaced with system radius server.

```
ACOS(config)# system radius server
ACOS(config-radius-server)# remote ip-list RADIUS_IP_LIST
ACOS(config-radius-server)# secret a10
ACOS(config-radius-server)# attribute inside-ipv6 vendor 22610 number
29
ACOS(config-radius-server)# attribute msisdn number 31
ACOS(config-radius-server)# attribute imei vendor 10415 number 20
ACOS(config-radius-server)# attribute imsi vendor 10415 number 1
ACOS(config-radius-server)# attribute custom1 l3v-c1 vendor 22610
number 42
ACOS(config-radius-server)# attribute custom2 l3v-c2 vendor 22610
number 43
ACOS(config-radius-server)# attribute custom3 cus3 vendor 22610 number
44
ACOS(config-radius-server)# attribute custom4 cus4 vendor 22610 number
81
```

```
ACOS(config-radius-server)# attribute custom5 cus5 vendor 22610 number
82
ACOS(config-radius-server)# attribute custom6 cus6 vendor 22610 number
83
ACOS(config-radius-server)# exit
```

6. Configure a RADIUS profile that assigns clients to the LSN LIDs based on the attribute values from the external RADIUS server. The following commands configure a CGN RADIUS profile:

```
ACOS(config)# cgnv6 lsn-radius-profile 1
ACOS(config-lsn-radius-rule)# radius custom1 exact-value l3v-c1 lsn-lid
2
ACOS(config-lsn-radius-rule)# radius default lsn-lid 1
ACOS(config-lsn-radius-rule)# exit
```

When you configure the class list of client IP addresses, you can map the client network to this profile.

The **radius** command matches on the attribute values from the external RADIUS server, and specifies the LSN LID to use for handling clients that have the matching attribute value.

Use the **custom1** to **custom6** option for the A10-CGN-Radius-Custom-1 to A10-CGN-Radius-Custom-6 attributes respectively. Use the **default**option for clients who do not have any of these custom attributes.

The commands in the following example assign clients to CGN NAT pools based on the RADIUS attribute mappings listed in Table 8

Table 8 : NAT Pool Assignments Based on RADIUS Attribute (values used in example below)

Attribute Name in Dictionary File	Attribute Name in ACOS	Mapped to this Name in ACOS	Assigns Clients to this LSN LID
A10-CGN-Radius-Custom-1	custom1	l3v-c1	LSN LID 1 Assigns clients to CGN NAT pool cgnpool1.
A10-CGN-Radius-	custom2	l3v-c2	LSN LID 2 Assigns clients to CGN

Table 8 : NAT Pool Assignments Based on RADIUS Attribute (values used in example below)

Attribute Name in Dictionary File	Attribute Name in ACOS	Mapped to this Name in ACOS	Assigns Clients to this LSN LID
Custom-2			NAT pool cgnpool2.
A10-CGN-Radius-Custom-3	custom3	cus3	LSN LID 3 Assigns clients to CGN NAT pool cgnpool3.
A10-CGN-Radius-Custom-4	custom4	cus4	LSN LID 4 Assigns clients to CGN NAT pool cgnpool4.
A10-CGN-Radius-Custom-5	custom5	cus5	LSN LID 5 Assigns clients to CGN NAT pool cgnpool5.
A10-CGN-Radius-Custom-6	custom6	cus6	LSN LID 6 Assigns clients to CGN NAT pool cgnpool6.
None of the above included in RADIUS Accounting Start record for client	No mapping. Uses the default LSN LID assignment in the RADIUS profile, if configured.	No mapping. Use the default LSN LID assignment, if configured.	LSN LID 7 Assigns clients to CGN NAT pool cgnpool7.

To match only on an exact attribute value, use the **`exact-value`** option followed by the portion of the attribute name on which to match. Similarly, to match based on only the beginning portion of an attribute value, use the **`starts-with`** option followed by only the portion of the attribute name on which to match. (See the “CLI Example” below.)

7. Map each attribute value expected from the external RADIUS server to a LSN LID.
8. Configure a class list that assigns clients to the RADIUS profile.

   The following commands configure the class list:

```
ACOS(config)# class-list CLASS_LIST
ACOS(config-class list)# 10.0.0.0/8 lsn-radius-profile 1
ACOS(config-class list)# 11.0.0.0/8 lsn-lid 1
ACOS(config-class list)# exit
```

NOTE: Based on the class list, clients with private addresses in the 10.x.x.x/8 subnet are handled based on CGN RADIUS profile 1. Clients with private addresses in the 11.x.x.x/8 subnet are handled based on LSN LID 1.

9. Set the class list as the list of client (inside) addresses for CGN.

```
ACOS(config)# cgnv6 lsn inside source class-list CLASS_LIST
```

10. To view the table of RADIUS attributes stored on the ACOS device, enter the following command:

```
ACOS(config)# show system radius table
LSN RADIUS Table Statistics:
----------------------------------------
Record Created                  1
Record Deleted                  0

MSISDN           IMEI               IMSI             Inside-IP
l3v-c1
l3v-c2
cus3
cus4
cus5
cus6
------------------------------------------------------------------------
-------
123456789012345  0000000002222222   000000000111111
2001:99:3301:133::1
cc1
```

```
Total RADIUS Records Shown: 1
```

The attribute names are listed, followed by the values received by the ACOS RADIUS server for these attributes for individual clients. NAT profile selection is based on the custom attribute names. In this example, the ACOS CGN configuration maps the RADIUS custom attributes to the following names:

- customl – Mapped to attribute name “l3v-cl”
- custom2 – Mapped to attribute name “l3v-c2”
- custom3 – Mapped to attribute name “cus3”
- custom4 – Mapped to attribute name “cus4”
- custom5 – Mapped to attribute name “cus5”
- custom6 – Mapped to attribute name “cus6”

Following the list of attribute names, the attribute values for individual clients are listed. This example shows information for a client. The information is from the sample Accounting Start record in Custom RADIUS Attributes. The client’s RADIUS information includes the custom attribute mapped LSN LID 1, so ACOS assigns the CGN NAT settings in that LSN LID to the client.

Multiple RADIUS Secret Keys Support

For users with multiple RADIUS systems, all of which use different secret keys, ACOS provides the ability to configure different secret keys in order to act with separate RADIUSM systems. Up to 8 RADIUS systems can be configured in an IP List within the RADIUS server configuration level. Each system can be configured with a separate secret key. Otherwise, a default secret key can be configured with the RADIUS server configuration level. All RADIUSM systems sharing that secret key must be specified in a single IP list.

Configuring RADIUS Secret Keys

To configure the RADIUS secret keys, enter the configuration level of an LSN RADIUS server by entering the following command at the global configuration level:

```
ACOS(config)# system radius server
```

To configure a default secret string, enter the following command at the RADIUS server configuration level:

```
ACOS(config)# system radius server
ACOS(config-radius-server)# remote ip-list RADIUS_IP_LIST1 secret a11
ACOS(config-radius-server)# remote ip-list RADIUS_IP_LIST2 secret a12
ACOS(config-radius-server)# remote ip-list RADIUS_IP_LIST3 secret a13
ACOS(config-radius-server)# remote ip-list RADIUS_IP_LIST4
ACOS(config-radius-server)# secret defaultsecretstring
ACOS(config-radius-server)# attribute inside-ip number 8
```

For more information about **system radius server** command, see *Command Line Interface Reference.*

Configuring Platform-based LSN RADIUS Table Size

To configure a custom LSN RADIUS table size, a new option has been added to the **cgnv6 resource-usage** command. The minimum and maximum configurable sizes will vary depending on the platform.

```
ACOS(config)# cgnv6 resource-usage radius-table-size 3000000
```

To view the current LSN RADIUS table size, as well as the default, maximum, and minimum values allowed for your platform, a new **radius-table-size** entry has been added to the **show cgnv6 resource-usage** command.

Configuring RADIUS Accounting Requests

In order to process RADIUS accounting-on messages, the RADIUS server must first be configured on the ACOS device. To configure table entry deletion upon receiving a RADIUS accounting-on request, you can configure an attribute to be used for deleting entries when accounting-on request is received or to ignore other incoming accounting-on requests.

1. To configure ACOS to delete entries matching a default attribute in the RADIUS table, enter the following command at the IP NAT LSN RADIUS server configuration level:

```
ACOS(config)# system radius server
```

```
ACOS(config-radius-server)# accounting on delete-entries-using-
attribute msisdn
```

2. To configure ACOS to delete entries matching a custom attribute, enter the following command at the RADIUS server configuration level:

```
ACOS(config)# system radius server
ACOS(config-radius-server)# accounting on delete-entries-using-
attribute attribute1
```

Disabling RADIUS Accounting Response

Whenever ACOS receives and successfully processes a RADIUS accounting request message, it sends a RADIUS accounting response in reply. If a confirmation is not needed, or if the user wants to limit the flood of response messages, then this option can be disabled so that no RADIUS accounting response is sent.

To disable RADIUS accounting responses from being sent in reply to RADIUS accounting requests, enter the following command at the `config-radius-server` configuration level:

```
ACOS(config)# system radius server
ACOS(config-radius-server)# disable-reply
```

By default, if a RADIUS server is configured on the ACOS device, then RADIUS accounting responses are sent when a RADIUS accounting request is processed successfully.

Framed IPv6 Prefix Support in RADIUS Table

While sending CGN logs, RADIUS attributes for a client can be added to the log messages. To achieve this, ACOS is configured to act as a RADIUS server so that it can receive RADIUS accounting requests that include the client RADIUS attributes.

When client's AAA server sends out RADIUS accounting packet that has the Framed IP and (/or) Framed IPv6 Prefix to ACOS, ACOS intercepts the packet, creates a RADIUS table entry based on the IP and IPv6 Prefix. When the inside user creates a data connection either from the IP or from IPv6 address (from the prefix), ACOS then includes the RADIUS attributes while sending the log messages.

ACOS acts as a RADIUS server intercepting RADIUS accounting request messages sent to the Interface / Floating IPs configured on ACOS. To create a RADIUS server configuration for CGNv6 deployment, use the **`system radius server`** command. For more information about **`system radius server`** command, see *Command Line Interface Reference*.

When configuring the CGNV6 RADIUS server, use the framed-ipv6-prefix command to specify the Framed IPv6 Prefix as a RADIUS attribute for RADIUS accounting requests. The following combination are possible in a RADIUS packet:

- Framed IPv4 address and Framed IPv6 prefix — ACOS accepts the packet and creates the RADIUS entries based on the IPv4 address and the IPv6 prefix.
- Framed IPv4 address and Framed IPv6 address — ACOS accepts the packet and create the RADIUS entries based on the IPv4 address and the IPv6 address.
- Framed IPv4 address — ACOS accepts the packet and creates the record with IPv4 address.
- Framed IPv6 address — ACOS accepts the packet and creates the record with IPv6 address.
- Framed IPv6 prefix — ACOS accepts the packet and creates the record with IPv6 prefix.

The Framed IPv6 prefix attribute in the RADIUS packet contains the prefix with the configured prefix length. When the configured prefix length on the RADIUS server does not match with the incoming prefix length, then the packet will be dropped.

When the prefix length is changed in the RADIUS server, the existing RADIUS table must be explicitly cleared.

NOTE:

- The value of the Framed IPv6 Prefix is configurable. If the configured prefix is changed, the RADIUS table must be explicitly cleared to remove the previously learned RADIUS table entries.
- ACOS accepts the RADIUS accounting packets only when the packet is destined to the ACOS Interface IP or Floating IP.

Configuration Example

The following configuration configures Framed IPv6 Prefix support for RADIUS table in CGN partition.

1. Enter the following command to create an IP list for client RADIUS servers:

```
ACOS(config)# ip-list RADIUS_IP_LIST
ACOS(config-ip list)# 9.9.9.9 to 9.9.9.10
ACOS(config-ip list)# exit
```

2. The following commands configure RADIUS server parameters for ACOS:

```
ACOS(config)# system radius server
ACOS(config-lsn radius)# remote ip-list RADIUS_IP_LIST
ACOS(config-lsn radius)# listen-port 1813
ACOS(config-lsn radius)# attribute inside-ip number 8
ACOS(config-lsn radius)# secret a10
ACOS(config-radius-server)# attribute inside-ip number 8
ACOS(config-radius-server)# attribute msisdn number 31
ACOS(config-radius-server)# attribute imei vendor 10415 number 20
ACOS(config-radius-server)# attribute imsi vendor 10415 number 1
ACOS(config-radius-server)# attribute custom1 l3v-c1 vendor 22610
number 42
ACOS(config-radius-server)# attribute custom2 l3v-c2 vendor 22610
number 43
ACOS(config-radius-server)# attribute custom3 cus3 vendor 22610 number
44
ACOS(config-radius-server)# attribute custom4 cus4 vendor 22610 number
81
ACOS(config-radius-server)# attribute custom5 cus5 vendor 22610 number
82
ACOS(config-radius-server)# attribute custom6 cus6 vendor 22610 number
83
ACOS(config-radius-server)# attribute inside-ipv6-prefix prefix-length
64 number 97
ACOS(config-radius-server)# attribute inside-ipv6 vendor 22610 number
29
ACOS(config-radius-server)# accounting start replace-entry
ACOS(config-radius-server)# accounting stop delete-entry-and-sessions
ACOS(config-radius-server)# accounting interim-update replace-entry
```

3. The following commands configure a logging template:

```
ACOS(config)# cgnv6 template logging log
ACOS(config-logging:log)# log http-requests url
ACOS(config-logging:log)# log sessions
ACOS(config-logging:log)# include-radius-attribute msisdn sessions
ACOS(config-logging:log)# include-radius-attribute imei sessions
ACOS(config-logging:log)# include-radius-attribute imsi sessions
ACOS(config-logging:log)# include-radius-attribute custom1 sessions
ACOS(config-logging:log)# include-radius-attribute custom2 sessions
ACOS(config-logging:log)# include-radius-attribute custom3 sessions
ACOS(config-logging:log)# include-radius-attribute custom4 sessions
ACOS(config-logging:log)# include-radius-attribute custom5 sessions
ACOS(config-logging:log)# include-radius-attribute custom6 sessions
ACOS(config-logging:log)# include-radius-attribute framed-ipv6-prefix
prefix-length 64
ACOS(config-logging:log)# include-http referer
ACOS(config-logging:log)# include-http user-agent
ACOS(config-logging:log)# include-http header1 GET
ACOS(config-logging:log)# include-http l4-session-info
ACOS(config-logging:log)# include-http method
ACOS(config-logging:log)# include-http request-number
ACOS(config-logging:log)# include-http file-extension
ACOS(config-logging:log)# rule http-requests dest-port 80
ACOS(config-logging:log)# rule http-requests log-every-http-request
ACOS(config-logging:log)# rule http-requests max-url-len 200
ACOS(config-logging:log)# rule http-requests include-all-headers
ACOS(config-logging:log)# rule http-requests disable-sequence-check
ACOS(config-logging:log)# batched-logging-disable
ACOS(config-logging:log)# service-group cgn-log-group
```

Configuring Hairpin Filter Matching for DS-Lite and NAT64

The hairpin filtering options also apply to DS-Lite and NAT64:

- For DS-Lite traffic, matching is based on the IPv6 source address and the tunneled IPv4 address.
- For NAT64, matching is based on the source IPv6 address.

You can enable one of the hairpin filtering options. The enabled option applies to LSN, DS-Lite, and NAT64 traffic.

Enter the following command to configure the filtering granularity for LSN hairpinning:

```
ACOS(config)# cgnv6 lsn hairpinning filter-none
```

This command is entered at the global configuration level of the CLI.

Configuring the LSN SYN Timeout

The LSN timeout ranges from 2 to 30 seconds, with a default setting of 4 seconds.

Enter the following command to change the LSN timeout:

```
ACOS(config)# cgnv6 lsn syn-timeout 30
```

Enabling or Disabling ALG Support in LSN

The following example enables ALG support for the ESP protocol in LSN:

```
ACOS(config)# cgnv6 lsn alg esp enable
```

The following example disables ALG support for the ESP protocol in LSN:

```
ACOS(config)# no cgnv6 lsn alg esp enable
```

Displaying ALG Information for LSN

To display ALG information for LSN:

- Enter the following command to display the state of LSN ALG support for ESP protocol:

  ```
  ACOS# show cgnv6 lsn alg esp config
  ```

- Enter the following command to display ALG statistics for a protocol in LSN:

  ```
  ACOS# show cgnv6 lsn alg esp statistics
  ```

Displaying information for GRE Sessions

Enter the following commands to display information for GRE sessions:

```
ACOS# show cgnv6 lsn user-quota-sessions
LSN User-Quota Sessions:
Inside Address NAT Address ICMP UDP TCP Session Pool LID Flag
------------------------------------------------------------------------
---------
10.10.10.4 19.19.19.103 0 1 0 1 lsn_p1 1 U
Total User-Quota Sessions Shown: 1
```

Configuring STUN Timeout

For SIP Contact NAT mappings, the corresponding full-cone session's Session Traversal Utilities for NAT (STUN) timeout is set to the *Expires* value in the SIP Registration packet's payload. For SIP RTP/RTCP NAT mappings, the corresponding full-cone session's STUN timeout is configurable. The RTP/RTCP STUN timeout can be 2-10 minutes, and the default is 5 minutes.

Enter the following command to change the RTP/RTCP STUN timeout for full-cone sessions used for SIP NAT mappings:

```
ACOS(config)# cgnv6 lsn alg sip rtp-stun-timeout 5
```

This command is entered at the global configuration level of the CLI.

Using NAT64 FTP PASV Mode with XLAT

You can use NAT64 passive mode (PASV) FTP to work with XLAT. You can also configure this option with the **`cgnv6 nat64 alg ftp xlat-no-trans-pasv enable`** command.

If you configure the option, and the client sends a PASV request, the PASV response from the server is not translated to an EPSV response. However, if an IPv6 client sends an EPSV request, FTP ALG translates the request to PASV. When the server responds, the message is translated from PASV to EPSV.

Disabling Port Preservation

By default, LSN attempts to use the same source protocol port for a client's public address (NAT address) that is used in the client's inside address. For example, if the client sends a request with the source port TCP 5000, LSN uses TCP 5000, if available, as the source port in the NATted request that is sent to the server.

This feature is called port preservation. If you disable port preservation, ACOS will not attempt to use the same protocol port in the client's inside address as the source protocol port in the client's public address.

NOTE: Even when port preservation is disabled, it is possible in rare cases for the same protocol port to be used.

Disabling Port Preservation

Enter the following command to disable or re-enable port preservation:

```
ACOS(config)# cgnv6 lsn attempt-port-preservation disable
```

If you disable port preservation after traffic has run through the ACOS device for some time, save the configuration by entering the **write memory** command and reload the device by entering the **reload** command.

Configuring TCP Maximum Segment Size Clamping

The TCP maximum segment size (MSS) specifies the maximum length, in bytes, of data that one SYN or SYN-ACK packet in a TCP connection can have. The MSS does not include the TCP or IP header.

MSS Clamping Methods

You can set TCP MSS clamping for LSN to be performed using one of the following methods:

- None – ACOS does not change the MSS value.
- Fixed value – ACOS changes the MSS to the specified length.
- Subtract – ACOS reduces the MSS if the value is greater than the specified number of bytes.

This option sets the MSS based on the following calculations (S - Value to subtract from the maximum MSS Clamping value and N - Minimum value of the MSS Clamping):

- If MSS minus S is greater than N, subtract S from the MSS.
- If MSS minus S is less than or equal to N, set the MSS to N.

  By default, the subtract method of MSS clamping is used with the following values:

  S = 40 bytes

  N = 416 bytes

  Using these values, the default MSS clamping calculations are as follows:

  - If MSS minus 40 is greater than 416, subtract 40 from the MSS.
  - If MSS minus 40 is less than or equal to 416, set the MSS to 416.

NOTE: TCP MSS clamping is supported with LSN One-to-One NAT.

Changing the MSS Clamping Method

Enter the following command to change the MSS clamping method for LSN to a fixed maximum value of 122:

```
ACOS(config)# cgnv6 lsn tcp mss-clamp fixed 122
```

Disabling TCP Resets in Response to Invalid TCP Packets

By default, if ACOS receives an invalid TCP packet from the inside network, ACOS sends a TCP reset for the host session. An invalid TCP packet is received without a matching session where a TCP RST will be sent. Optionally, you can disable TCP resets from being sent in this situation.

Enter following command to disable TCP resets in response to invalid TCP packet from the inside network:

```
ACOS(config)# cgnv6 lsn tcp reset-on-error outbound disable
```

Configuring ICMP Options

You can configure the following ICMP/ICMPv6 options for IPv6 migration:

Destination Unreachable Options

ACOS can send ICMP Unreachable messages when one of the following conditions have been met:

- A configured user quota has been exceeded.
- No NAT ports are available for mappings.

By default, ACOS sends code type 3, code 13, administratively filtered, when a configured user quota is exceeded. ICMP Unreachable messages when no NAT ports are available for mappings is disabled by default.

Enter the following command to change the behavior:

```
ACOS(config)# cgnv6 lsn icmp  send-on-port-unavailable admin-filtered
```

Configuring Ping Replies from NAT Pool Addresses

Ping Replies from NAT Pool Addresses by Using the GUI

You cannot configure this option in the GUI.

Ping Replies from NAT Pool Addresses by Using the CLI

Enter the following commands to enable ping replies from NAT pool addresses:

```
ACOS(config)# cgnv6 nat icmp respond-to-ping
ACOS(config)# cgnv6 nat icmpv6 respond-to-ping
```

Configuring Source NAT for ICMP Error Messages

Configuring Source NAT for ICMP Error Messages by Using the GUI

You cannot configure this option by using the GUI.

Configuring Source NAT for ICMP Error Messages by Using the CLI

Enter the following command to enable NAT for ICMP messages from inside routers:

```
ACOS(config)# cgnv6 nat icmp always-source-nat-errors
```

This command is entered at the global configuration level of the CLI.

Configuring LSN Support for SCTP

To enable support for SCTP, configure an IP NAT inside and an IP NAT outside on the desired interfaces. Once the IP NAT inside and outside are configured, configure Static NAT.

Configuring Source and Destination NAT Support for SCTP

When a SYN packet is received, ACOS checks if NAT should be performed on both the source and destination IP. When the packet is received on the NAT_INSIDE interface and is destined to NAT_IP, source and destination NAT is performed. Since destination NAT is performed, the outgoing interface must also have the NAT_INSIDE interface configured. NAT_INSIDE should be configured on both the Rx and Tx interface. To configure source and destination NAT support, enter the following command:

```
ACOS(config)# cgnv6 nat inside source static 2.2.2.2 192.1.1.0
ACOS(config)# cgnv6 nat inside source static 10.1.1.1 192.11.1.1
```

For hair-pinning, inside user and outside user can be used interchangeably. Typically, the user initiating the connection is considered the inside user.

Configuring SCTP Timeout

The half-open timeout for SCTP traffic is configured in seconds. To configure a half-open timeout for SCTP traffic, enter the following command in the CLI:

```
ACOS(config)# cgnv6 sctp half-open-timeout 4
```

The idle timeout for SCTP traffic is configured in minutes. To configure an idle timeout for SCTP traffic, enter the following command in the CLI:

```
ACOS(config)# cgnv6 sctp idle-timeout 4
```

Configuring SCTP Restrictions

To restrict to only H.323 protocols is allowed in the SCTP DATA chunks, enter the following command in the CLI:

```
ACOS(config)# cgnv6 sctp permit-payload-protocol h.323
```

To configure packet rate-limiting for SCTP sessions, enter the following command in the CLI:

```
ACOS(config)# cgnv6 sctp rate-limit source 1.1.1.1
```

The commands to configure payload protocols and SCTP packet rate-limiting are only available on CFW platforms.

Modifying LSN NAT Pool without Downtime

When sessions are running, you can edit or modify the NAT pool without the need to clear the sessions first. When the NAT pool has been modified, the current session is kept active on the old pool in the background until the sessions end. New sessions are mapped to the new NAT pool using new NAT addresses. When all sessions using old NAT addresses end, ACOS releases the old NAT addresses from the system.

If the public NAT IP is distributed using a routing protocol (for example, BGP), ACOS stops redistributing the old public IP address until all the sessions using this public IP has been cleared first in the background. New public IP addresses are redistributed immediately when the NAT pool has been modified.

To view the status of old NAT address, enter the following command:

```
ACOS# show cgnv6 nat pool statistics brief
LSN Address Pool Statistics:
-------------------------------------------------------------------------------
-----------
test             Address          Users     UDP       TCP
-------------------------------------------------------------------------------
------------
                  80.1.1.6         112       112       0
                  80.1.1.7         101       101       0
                  80.1.1.8         72        72        0
                  80.1.1.9         72        72        0
                  80.1.1.10        72        72        0
```

```
(obsoleted)       80.1.1.5          2         2         0
(obsoleted)       80.1.1.4          2         2         0
(obsoleted)       80.1.1.3          2         2         0
(obsoleted)       80.1.1.2          2         2         0
(obsoleted)       80.1.1.1          2         2         0
```

Once the old addresses are in the obsoleted status, they will be kept in the system until all sessions using them end.

NOTE: You can modify the NAT address range repeatedly only when all obsoleted address from the last change have been completely removed from the system.

```
ACOS(config)# show cgnv6 nat pool-group grp1 statistics
NAT Pool Group Statistics
-------------------------
Pool Group Name      Total IP   Used IP    Free IP
-----------------------------------------------------
grp1                 145        0          145

ACOS(config)# show cgnv6 nat pool-group statistics
NAT Pool Group Statistics
-------------------------
Pool Group Name      Total IP   Used IP    Free IP
-----------------------------------------------------
grp2                 120        0          120
grp1                 145        0    To    145
-----------------------------------------------------
```

For information, see the *Command Line Interface Reference*.

Deploying CGN with L3V Inter-partition Routing

L3V Statistics

CGN statistics for L3V displays information on counters for each partition by using the supported show commands. NAT pool usage statistics are displayed in the partition that owns the NAT pool.

The **axdebug** debug and monitoring command displays session-specific information for each partition.

For inter-partition traffic, the following types of IPv6 Migration technology statistics are collected in the partition to which the private client belongs:

- **show cgnv6 lsn statistics**
- **show cgnv6 ds-lite statistics**
- **show cgnv6 nat64 statistics**
- **show cgnv6 fixed-nat statistics**

The aggregate packet level statistics are collected in the partition that receives the packet. You can display the output of, for example, the **show slb switch** and **show** {**ip** | **ipv6**} **fragmentation statistics** commands.

Consider the following information:

- Using overlapping IP address spaces in private partitions is supported but is not required.
- VLAN and VE IDs must be unique across a given ACOS device.
- The same VLAN/VE ID cannot be used in more than one partition.
- L3V also is supported with Fixed-NAT.
- For more information, see Configuring L3V Inter-partition Routing for Fixed-NAT.
- For information about CGN logging with L3V, see the Traffic Logging Guide for IPv6 Migration.
- For more information about L3V, see the System Configuration and Administration Guide.

Configuring ADP for CGN

The commands in this example configure the L3V CGN deployment shown in this figure.

Configuring the Partition p0

The commands in this section configure partition “p0”. To begin, the following commands change the CLI to the partition:

```
ACOS-Active(config)# end
ACOS-Active# active-partition p0
```

```
Currently active partition: p0
```

The following commands configure the interface to internal clients, and enable inside NAT on the interface:

```
ACOS-Active[p0]# configure
ACOS-Active[p0](config)# vlan 100
ACOS-Active[p0](config-vlan:100)# tagged ethernet 1
ACOS-Active[p0](config-vlan:100)# router-interface ve 100
ACOS-Active[p0](config-vlan:100)# interface ve 100
ACOS-Active[p0](config-if:ve100)# ip address 10.10.10.1 255.255.255.0
ACOS-Active[p0](config-if:ve100)# ip nat inside
ACOS-Active[p0](config-if:ve100)# exit
```

The following command configures a static IP route from the private partition to the shared partition. The **partition shared** option specifies that the next hop for the route is the shared partition.

```
ACOS-Active[p0](config)# ip route 0.0.0.0 /0 partition shared
```

The following command enables VRRP-A for the partition. VRID 1 is configured to base its Active/Standby state on the state of VRID lead “leader”, configured in the shared partition:

```
ACOS-Active[p0](config)# vrrp-a vrid 1
ACOS-Active[p0](cofngi-vrid:1)# follow vrid-lead vrid1-leader
```

The following commands configure the CGN LID:

```
ACOS-Active[p0](config)# cgnv6 lsn-lid 1
ACOS-Active[p0](config-lsn lid)# source-nat-pool pool0 shared
```

The **source-nat-pool** option configures the private partition to use pool “pool0” in the shared partition to obtain NAT addresses for client mappings.

The following commands configure the class list:

```
ACOS-Active[p0](config-lsn lid)# class-list lsn
ACOS-Active[p0](config-class list)# 0.0.0.0/0 lsn-lid 1
ACOS-Active[p0](config-class list)# end
```

The following command activates CGN:

```
ACOS-Active[p0](config-class list)# cgnv6 lsn inside source class-list lsn
```

Configuring Partition p1

The following commands configure partition “p1”. The configuration is similar to the one for partition “p0”, except for the VLAN and accompanying VE ID. VLAN and VE IDs are required to be unique across all partitions of the device. A VLAN’s VE ID must be the same as that VLAN’s VE ID.

IP addresses must be unique in a partition but the same IP addresses can be used in more than one partition. In this example, both private partitions use the same address space for inside clients.

```
ACOS-Active# active-partition p1
Currently active partition: p1
ACOS-Active[p1]# configure
ACOS-Active[p1](config)# vlan 101
ACOS-Active[p1](config-vlan:101)# tagged ethernet 1
ACOS-Active[p1](config-vlan:101)# router-interface ve 101
ACOS-Active[p1](config-vlan:101)# interface ve 101
ACOS-Active[p1](config-if:ve101)# ip address 10.10.10.1 255.255.255.0
ACOS-Active[p1](config-if:ve101)# ip nat inside
ACOS-Active[p1](config-if:ve101)# exit
ACOS-Active[p1](config)# ip route 0.0.0.0 /0 partition shared
ACOS-Active[p1](config)# vrrp-a vrid 1 follow vrid-lead vrid1-leader
ACOS-Active[p1](config) class-list lsn
ACOS-Active[p1](config-class list)# 0.0.0.0/0 lsn-lid 1
ACOS-Active[p1](config-class list)# cgnv6 lsn inside source class-list lsn
ACOS-Active[p1](config)# cgnv6 lsn-lid 1
ACOS-Active[p1](config-lsn lid)# source-nat-pool pool0 shared
ACOS-Active[p1](config-lsn lid)# end
```

Configuring the Shared Partition

The following commands configure an Ethernet interface to the Internet and enable outside NAT on the interface:

```
ACOS(config)# vlan 200
ACOS(config-vlan:200)# tagged ethernet 2
ACOS(config-vlan:200)# router-interface ve 200
ACOS(config-vlan:200)# interface ve 200
ACOS(config-if:ve200)# ip address 172.7.7.1 255.255.255.0
ACOS(config-if:ve200)# ip nat outside
ACOS(config-if:ve200)# exit
```

The following commands configure VRRP-A for the shared partition:

```
ACOS(config)# vrrp-a common
ACOS(config-common)# device-id 1
ACOS(config-common)# set-id 1
ACOS(config-common)# enable
ACOS(config-common)# vrrp-a vrid 1
ACOS-Active(config-vrid:1)# vrrp-a vrid-lead vrid1-leader
ACOS(config-vrid-lead:vrid1-leader)# partition shared vrid 1
ACOS(config-vrid-lead:vrid1-leader)# exit
```

The **vrrp-a vrid-lead** option configures a VRRP-A lead. Later in the configuration, the partitions are configured to follow the shared partition's VRID state. When the shared partition's VRID is active, the VRID of each of the private partitions that follows the shared partition's VRID state also become active. If the shared partition's VRID state changes to Standby, the VRID state of each of the private partition VRIDs also change Standby.

The following commands create the L3V partitions:

```
ACOS-Active(config)# partition p0 id 1 application-type cgnv6
ACOS-Active(config)# partition p1 id 2 application-type cgnv6
```

The following commands configure a partition group containing the L3V partitions:

```
ACOS-Active(config)# partition-group pg0
ACOS-Active(config-config-partition-group:pg0)# member p0
ACOS-Active(config-config-partition-group:pg0)# member p1
ACOS-Active(config-config-partition-group:pg0)# exit
ACOS-Active(config)#
```

The following command configures an LSN NAT pool to be shared by the private partitions:

```
ACOS-Active(config)# cgnv6 nat pool pool0 1.1.1.1 1.1.1.100 netmask /24
vrid 1 shared group pg0
```

The **vrid 1 shared** option adds the pool addresses to the shared partition's VRID for backup. The **group** option shares the pool with the CGN configurations in the private partitions in group "pg0".

Configuring Destination NAT Port Translation

To configure a port-list, use the following commands:

```
ACOS(config)# cgnv6 port-list abc
ACOS(config-port-list)# original-port 80 to translated-port 8080
ACOS(config-port-list)# original-port 5353 to translated-port 53
```

To bind the port-list to an lsn-rule-list, use the following commands:

```
ACOS(config)# cgnv6 lsn-rule-list r1
ACOS(config-lsn-rule-list)# default
ACOS(config-lsn-rule-list-default)# tcp port 1 to 65535 action dnat ipv4-list i port-list abc
ACOS(config-lsn-rule-list-default)# udp port 1 to 65535 action dnat ipv4-list i port-list abc
```

NOTE: Port-list can be configured only if there is an ipv4-list configured for dnat.

To view the information of configured port-lists, use the **show running-config** command.

Displaying and Clearing LSN Information

The following section displays the LSN information.

- Enter the following command to display the configured class lists:

```
ACOS# show class-list list1
Total single IP:     0
Total IP subnet:     1
Content:
    192.168.0.0 /16 lsn-lid 5
```

- Enter the following command to display the currently active full-cone sessions:

```
ACOS# show cgnv6 lsn full-cone-sessions
LSN Full Cone Sessions:
Prot Inside Address         NAT Address        Conns  Pool    CPU  Age
```

```
-----------------------------------------------------------------------
TCP  192.168.1.1:20001      203.0.113.1:20001   1     pool1    1    0
TCP  192.168.2.1:30001      203.0.113.1:30001   1     pool1    4    0
TCP  192.168.255.1:50001    203.0.113.1:50001   1     pool1    13   0
```

- Enter the following command to display the currently active user quota sessions:

```
ACOS# show cgnv6 lsn user-quota-sessions
LSN User-Quota Sessions:
Inside Address       NAT Address          ICMP  UDP  TCP  Session Pool
LID
-------------------------------------------------------------------------
---
192.168.1.1:20001    203.0.113.1:20001    0     3    0    0       pool1
3
192.168.2.1:30001    203.0.113.1:30001    0     3    0    0       pool1
3
192.168.255.1:50002  203.0.113.1:50001    0     2    0    0       pool1
3
```

- Enter the following command to display the configured LSN static port reservations:

```
ACOS# show cgnv6 lsn port-reservations
LSN Port Reservations
Inside Address                    Start    End      NAT Address
Start    End
-------------------------------------------------------------------------
--------------
192.168.1.1                       80       1024     203.0.113.1      80
    1024
```

- Enter the following command to display the system-level information for LSN:

```
ACOS(config)# show cgnv6 lsn system-status
CPU Usage:
----------
Control CPU 1 :   6%
Data CPU 1    :   0%
Data CPU 2    :   0%
Data CPU 3    :   0%
```

```
Data CPU 4     :    0%
Data CPU 5     :    0%
Data CPU 6     :    0%
Data CPU 7     :    0%
Data CPU avg   :    0%

Memory Status:
--------------
Total Memory(KB): 12291491
Used Memory(KB) : 8814491
Free Memory(KB) : 3477000
Memory Usage    : 71.7%

Sessions Status:
----------------
LSN CPS           : 0
Data Sessions Used: 0
Data Sessions Free: 33456123
SMP Sessions Used : 0
SMP Sessions Free : 33292288

NAT Port Usage:
---------------
TCP NAT Ports Used: 0
TCP NAT Ports Free: 17095680
UDP NAT Ports Used: 0
UDP NAT Ports Free: 17095680

RADIUS Table Usage:
-------------------
RADIUS Entries Used: 0
RADIUS Entries Free: 1500000
```

- Enter the following command to display the global statistics related to LSN:

```
ACOS(config)# show cgnv6 lsn statistics
```

- Enter the following command to display the LSN ALG information:

```
ACOS(config)# show cgnv6 lsn algprotocol
```

- Enter the following command to display the current and configurable values for

system resources:

```
ACOS# show system resource-usage
Resource                          Current    Default    Minimum
Maximum
------------------------------------------------------------------------
--
l4-session-count                  67108864   67108864   16777216
134217728
class-list-ipv6-addr-count        4096000    4096000    4096000
8192000
class-list-ac-entry-count         3072000    3072000    3072000
6144000
auth-portal-html-file-size        20         20         4          120
auth-portal-image-file-size       6          6          1          80
max-aflex-file-size               32768      32768      16384      262144
aflex-table-entry-count           102400     102400     102400
10485760
```

The **Current** column shows the maximum number of LSN pool addresses that are currently allowed on the system. The **Default** column displays the allowable maximum value. In this example, the administrator increased the maximum value to 10000.

The maximum value can be any value in the range between the values in the Minimum and Maximum columns in the output.

NOTE: If you change the maximum number of Layer 4 sessions (l4-session-count), a reload will not place this change into effect. You must reboot the device.

- Enter the following command to display the configured NAT pools:

```
ACOS# show cgnv6 nat pool
Total IP NAT Pools: 3
Pool Name     Start Address End Address  Mask  Gateway  Vrid
-------------------------------------------------------------
test-lsn-pool 100.101.1.1   100.101.1.12  /24  0.0.0.0  default
```

- Enter the following command to display NAT pool statistics:

```
ACOS(config)# show cgnv6 nat pool
LSN Address Pool Statistics:
----------------------------
Pool Name Total IPs Total Users Free IPs Used IPs UDP-port-overloaded
TCP-port-overloaded
------------------------------------------------------------------------
------------------
pool1 3 4 0 3 5 0
pool1 Address Users ICMP Freed Total UDP Freed Total Rsvd oLoaded TCP
Freed Total Rsvd oLoaded
------------------------------------------------------------------------
------------------------
203.0.113.1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
203.0.113.2 2 0 0 0 5 0 0 0 5 0 0 0 0 0
203.0.113.3 1 0 0 0 0 0 0 0 0 0 0 0 0 0
```

- Enter the following command to display the NAT pool group statistics:

```
ACOS(config)# show cgnv6 nat pool-group statistics
NAT Pool Group Statistics
-------------------------
Pool Group Name      Total IP   Used IP    Free IP
-------------------------------------------------
grp2                 120        0          120
grp1                 145        0   To   145
-------------------------------------------------
```

- Enter the following command to display the counters for the NAT pool:

```
ACOS(config)# show counters cgnv6 nat pool
/cgnv6/nat/pool/pool1
**************************************
Users                                                              1
ICMP                                                               0
ICMP Freed                                                         0
ICMP Total                                                         0
ICMP Reserved                                                      0
ICMP Peak                                                          0
ICMP Hit Full                                                      0
UDP                                                                0
UDP Freed                                                          0
```

```
UDP Total                                                    0
UDP Reserved                                                 64507
UDP Peak                                                     0
UDP Hit Full                                                 0
UDP Port Overloaded                                          0
UDP Port Overloading Session Created                         0
UDP Port Overloading Session Freed                           0
TCP                                                          0
TCP Freed                                                    0
TCP total                                                    0
TCP Reserved                                                 64507
TCP Peak                                                     0
TCP Hit Full                                                 0
TCP Port Overloaded                                          0
TCP Port Overloading Session Created                         0
TCP Port Overloading Session Freed                           0
IP Used                                                      1
IP Free                                                      0
IP Total                                                     1
```

To clear LSN statistics or sessions:

- Enter the following commands to clear LSN statistics:

```
ACOS(config)# clear cgnv6 lsn statistics
ACOS(config)# clear cgnv6 lsn alg esp statistics
```

- Enter the following commands to clear LSN sessions:

```
ACOS(config)# clear cgnv6 lsn full-cone-sessions
ACOS(config)# clear cgnv6 lsn data-sessions
ACOS(config)# clear cgnv6 lsn all-sessions pool  p1
```

NOTE: Enter the final line of text before you remove a pool from a pool group.

- Enter the following command to clear ALG statistics for LSN:

```
ACOS(config)# clear cgnv6 lsn alg espstatistics
```

The **rtsp** option clears all ALG statistic counters, except the Current ALG sessions and Current Port mappings counters.

For detailed information about the show commands and the counters, see *Command Line Reference*.

NAT64 / DNS64

This chapter provides information about how to configure NAT64 and DNS64. It also provides information about additional configuration options.

The following topics are covered:

Overview

- NAT64/DNS64 is based on the following RFCs:
  - RFC 6146, *Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers*
  - RFC 6147, *DNS64: DNS Extensions for Network Address Translation from IPv6 Clients to IPv4 Servers*
- Using LSN and NAT64/DNS64 on the same ACOS device is supported.
- LSN and NAT64 can use the same NAT pool(s).

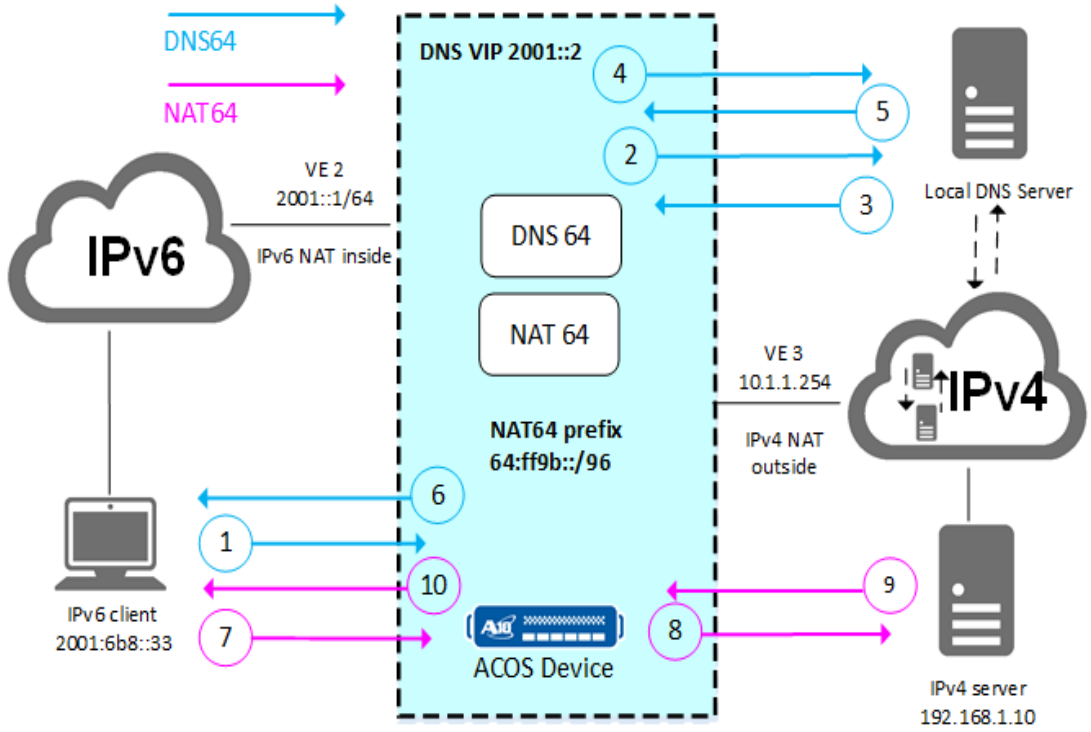
  For information about matching and traffic handling based on destination, see Destination Based NAT.

  For information about configuring user quotas by prefix, see User Quotas Based on IPv6 Prefix.
- For information about logging, see the Traffic Logging Guide for IPv6 Migration.
- Fixed-NAT is a log optimization feature that allocates NAT ports for each client from a predetermined (“fixed”) set of ports on the NAT address, without the need for logging.

  For information, see Fixed-NAT.
- In order to prevent translated IPv4 packets from being blocked during IPv4 identification checking on security devices, a non-zero identification field in the IPv4 packet header can be set if there is no IPv6 fragment header This feature enhances NAT64 translation. This feature applies to packet sizes greater than 88 bytes and less than or equal to 1280 bytes. The `all` option enables this behavior for packets of all sizes. This feature is disabled by default.

DNS64 and NAT64 work together to help IPv6 clients communicate with IPv4 servers. Figure 14 shows how DNS64 and NAT64 on an ACOS device help an IPv6 client establish a TCP session with an IPv4 server.

Figure 14 : DNS64 and NAT64



DNS64 on the ACOS device intercepts the client's IPv6 DNS request on the DNS VIP, and DNS64 forwards the AAAA request on behalf of the client. If the request results in a reply with an empty ANSWER section, or an error, or no reply, DNS64 sends an IPv4 DNS request instead.

The ACOS device retrieves the IPv4 addresses from the ANSWER section in the A reply and synthesizes an AAAA reply by changing the IPv4 addresses in the ANSWER section into IPv6 addresses.

The IPv6 addresses in the synthesized reply are constructed as follows (assuming a /96 prefix):

NAT64-prefix::*hex-version-of-IPv4-addr*

The following complementary features enable IPv6 clients to access IPv4 servers:

- DNS64 – Performs IPv4 and IPv6 DNS queries on behalf of IPv6 clients, and synthesizes IPv6 replies based on the IPv4 replies as required.

- NAT64 – Translates IPv6 packets from clients into IPv4 packets for communication with IPv4 servers. Likewise, NAT64 translates the IPv4 packets in server replies into IPv6 packets to send to the client.

For information about how the NAT64 prefix is used by DNS64, see NAT64 Prefix.

One-to-One NAT Support

One-to-One NAT supports both NAT44 and NAT64. ACOS provides support for the One-to-One NAT mappings based on the destination IP address. When an inside client connects to a server, ACOS creates a One-to-One NAT mapping with a bidirectional NAT, which allows the outside clients to connect to any port on the inside client.

For traffic from the inside client to a destination other than the server, ACOS will continue to use the normal, dynamic NAT.

The maximum supported One-to-One NAT IPs vary based on the platform memory. For more information about One-to-One NAT support and the maximum supported NAT IPs for the different platform memories, see One-to-One NAT Based on the Destination IP.

You can enable logging for one-to-one NAT64 using the following commands:

```
ACOS(config)#cgnv6 template logging log_template_name
ACOS(config-logging:log_template_nam)#log one-to-one-nat sessions
```

The log formats supported are CEF and ASCII.

For enabling One-to-One NAT64 logging and viewing the log samples, see *Traffic Logging Guide*.

TCP Sessions with an IPv4 Server

You can establish TCP sessions with an IPv4 server.

NOTE: This example assumes that the default DNS64 settings in ACOS are used.

This figure illustrates the following steps:

1. The IPv6 client 2001:6b8::33 sends a DNS request for the IPv6 address for www.example.com.

   The site resides on an IPv4 server.

2. DNS64 begins by sending an AAAA (IPv6 address record) query for the server's IPv6 address.
3. Since the IPv4 server does not have an IPv6 address, the DNS query results in an empty answer or an error.
4. When DNS64 receives the empty response or error, DNS64 sends an A (IPv4 address) record query for www.example.com's IPv4 address.
5. IPv4 DNS replies with the site's IPv4 address, 192.168.1.10.
6. DNS64 synthesizes an AAAA reply, which lists the IPv4 server's IP address, 64:ff9b::c0a8:10a.

   This is the server's IPv4 address converted into hexadecimal, and appended to the NAT64 prefix (64:ff9b::/96).

7. The client sends an IPv6 TCP SYN to 64:ff9b::c0a8:10a.
8. NAT64 creates a NAT session for the client, which replaces the client's IPv6 address with an IPv4 address from the NAT pool.

   NAT64 also replaces the IPv6 destination address with the corresponding IPv4 address of the server.

9. The ACOS device forwards the NAT IPv4 TCP SYN to the server.

   The TCP SYN has source IP address 10.1.1.1 and destination address 192.168.1.10.

10. The server replies with an IPv4 SYN-ACK.

    The SYN-ACK has the source IP address 192.168.1.10 and destination address 10.1.1.1.

11. The ACOS device translates the SYN-ACK into an IPv6 SYN-ACK and forwards it to the client.

    The SYN-ACK has the source IP address 64:ff9b::c0a8:10a and the destination address 2001:6b8::33.

Synthesis of AAAA Replies

Here is an example of a DNS reply from an IPv4 DNS server for an IPv4 query. The ANSWER section is highlighted.

```
; <<>> DiG 9.5.0b2 <<>> @3142::200 www.l.example.com
; (1 server found)
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52089
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 4, ADDITIONAL: 4

;; QUESTION SECTION:
;www.l.example.com.             IN      A

;; ANSWER SECTION:
www.l.example.com.      173     IN      A       192.168.1.10
www.l.example.com.      173     IN      A       192.168.1.11
www.l.example.com.      173     IN      A       192.168.1.12
www.l.example.com.      173     IN      A       192.168.1.13
www.l.example.com.      173     IN      A       192.168.1.14

;; AUTHORITY SECTION:
example.com.            68814   IN      NS      ns3.example.com.
example.com.            68814   IN      NS      ns1.example.com.
example.com.            68814   IN      NS      ns2.example.com.
example.com.            68814   IN      NS      ns4.example.com.

;; ADDITIONAL SECTION:
ns1.example.com.                168132  IN      A       172.16.1.10
ns2.example.com.                168132  IN      A       172.16.2.10
ns3.example.com.                168132  IN      A       172.16.3.10
ns4.example.com.                168132  IN      A       172.16.4.10

;; Query time: 5 msec
;; SERVER: 3142::200#53(3142::200)
;; WHEN: Wed Feb  2 13:40:55 2011
;; MSG SIZE  rcvd: 250
```

Here is an example of a AAAA reply that is synthesized by DNS64. DNS64 replaces the IPv4 addresses in the ANSWER section with IPv6 addresses. Each synthesized IPv6 address is a combination of the NAT64 prefix and the hexadecimal version of the IPv4 address.

```
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 3314
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 4, ADDITIONAL: 4

;; QUESTION SECTION:
;www.l.example.com.              IN      AAAA

;; ANSWER SECTION:
www.l.example.com.      147     IN      AAAA    64:ff9b::c0a8:10a
www.l.example.com.      147     IN      AAAA    64:ff9b::c0a8:10b
www.l.example.com.      147     IN      AAAA    64:ff9b::c0a8:10c
www.l.example.com.      147     IN      AAAA    64:ff9b::c0a8:10d
www.l.example.com.      147     IN      AAAA    64:ff9b::c0a8:10e

;; AUTHORITY SECTION:
example.com.            74649   IN      NS      ns2.example.com.
example.com.            74649   IN      NS      ns4.example.com.
example.com.            74649   IN      NS      ns1.example.com.
example.com.            74649   IN      NS      ns3.example.com.

;; ADDITIONAL SECTION:
ns1.example.com.                168132  IN      A       172.16.1.10
ns2.example.com.                168132  IN      A       172.16.2.10
ns3.example.com.                168132  IN      A       172.16.3.10
ns4.example.com.                168132  IN      A       172.16.4.10

;; Query time: 1 msec
;; SERVER: 3142::200#53(3142::200)
;; WHEN: Wed Feb  2 12:03:40 2011
;; MSG SIZE  rcvd: 326
```

NAT64 Prefix

The NAT64 prefix portion of the IPv6 address in the ANSWER section and the routes along the network path between the client and the ACOS device ensures that the client's IPv6 request for the site is handled by NAT64 on the ACOS device. NAT64 knows the server's IPv4 address from the portion of the synthesized IPv6 address that contains the IPv4 server's address

In this example, the NAT64 prefix is 64:ff9b::/96, the well-known prefix for NAT64 and DNS64. On the ACOS device, the prefix is a configurable NAT64 option. The prefix setting applies to NAT64 and DNS64. For syntax information, see Configuring NAT64.

Support for Multiple NAT64 Prefixes

You can set up to 64 NAT64 prefixes on an ACOS device, and configure prefix binding to a specific class-list. This is useful when you partition IPv6 users into different networks and manage these users separately.

This feature extends the functionality of NAT64 prefixes. The syntax to configure a NAT64 prefix is similar to previous releases and no configuration changes are required to enable enhancement.

NOTE: You can configure only one default NAT64 prefix in a class-list.

Support for NAT64 Prefix VRRP Active-Active VRID

You can use the `follow-nat-pool-vrid` option to support active-active VRRP configurations in NAT64 deployments. When the `follow-nat-pool-vrid` option is enabled on a NAT64 prefix, the traffic matching the NAT64 prefix will use the VRID of the NAT pool. Based on the NAT64 class-list configuration, multiple NAT pools with distinct VRIDs can be assigned, enabling active-active VRRP behavior.

Once a NAT64 prefix is configured with `follow-nat-pool-vrid`, route advertisement is disabled for that prefix. Traffic must be routed to the floating IP using Policy-Based Forwarding (PBF). The PBF configuration must match with the NAT64 configuration on ACOS to ensure accurate traffic forwarding.

DNS Template Options for DNS64

You can configure the following DNS64 options in a DNS template:

- **`Answer-only`** – DNS64 synthesizes only the resource records in the ANSWER section.

  For more information, see Synthesis of AAAA Replies. This option is enabled by default. If you disable this option, the IPv4 addresses in all other sections are synthesized to IPv6 too.

- **`Auth-data`** – When ACOS receives an A-query-response from the DNS server, it sets the authenticated-data bit in the synthesized AAAA response.

  The auth-bit is set only if DNS64 synthesis is performed in the reply. Otherwise, the bit is not changed. By default, this option is disabled.

- **`Cache`** – The ACOS device uses a cached A-query response to provide AAAA query responses for the same hostname without consulting the DNS server.

  For example, an A query has been cached for hostname example.com. If the client sends an AAAA query for example.com, ACOS does not consult the DNS server. Instead, ACOS uses the cached type A answer to synthesize an AAAA response and sends the synthesized response to the client. By default, this option is disabled.

- **`DNS64 state`** – Enabled or disabled.

- **`Change-query`** – When ACOS receives an AAAA request from a client, ACOS forwards only an A request on behalf of the client.

  This option saves time if the DNS database only contains A records, because ACOS does not need to wait for an error, an empty response, or for the response to time out. By default, the change-query option is disabled.

- **`Compress`** – To save network costs, in the DNS protocol, the DNS packet can be compressed.

  For example, www.example.com may occur many times in the DNS packet. For the first occurrence, ACOS uses the fully-qualified domain name (FQDN), which is 16 bytes long. The remaining occurrences can be displayed as an offset from the DNS

header (2 bytes), which saves 14 bytes for each subsequent occurrence of the name. By default, ACOS compresses each packet.

If you disable this option, ACOS will not compresses the packets. Even after disabling compression, if the name is the same as the FQDN in the QUESTION record, the packet is compressed without any performance cost.

- **`Deep-check-RR`** – Certain DNS64 requirements may need DNS64 to step through the resource records in the ANSWER section and apply certain rules.

  For example, the drop-CNAME option requires ACOS to evaluate the resource records individually. In this case, it is required to enable deep-check-RR along with drop-CNAME.

- **`Drop-CNAME`** – Sometimes the DNS server might send only CNAMEs in the ANSWER section in response to an AAAA query.

  This option drops these responses, considers the responses to be empty, and initiates an A query towards the hostname. By default, this option is enabled. This option is valid only when the deep-check-RR option is enabled.

NOTE: The drop-CNAME option is available only as a suboption of the deep-check-RR option.

- **`Ignore-rcode3`** – The ACOS device ignores a DNS response with rcode 3 in response to a AAAA query.

  The ACOS device treats the response as empty, and sends an A query to the same hostname. This option is useful for circumventing DNS servers that are configured incorrectly to return rcode=3 when they do not have any AAAA records for the hostname, even though the hostname exists. By default, this option is enabled.

- **`Max-qr-length`** – If the question-record length is greater than this value, the response from the DNS server is forwarded to the client without any modification to the response.

  You can specify 1-1023 bytes, and the default is 128.

- **`Parallel-query`** – The ACOS device sends an IPv6 AAAA request and an IPv4 A request in parallel on behalf of the client.

By default, this option is disabled. When this option is enabled, ACOS performs DNS64 synthesis and forwards the first valid response that is received to the client. Empty responses and errors are invalid.

If both responses are invalid, ACOS forwards the last invalid response to the client.

NOTE: It is recommended to disable the passive-query option and enable the single-response option when using the parallel-query option.

- **`Passive-Query`** – When an empty response or an AAAA query is received, ACOS initiates an A query. By default, this option is enabled.
- **`Retry`** – When this option is enabled, and a response is not received from the DNS server, ACOS retries an A query. By default, ACOS will retry 3 times for each query received from the server. This option can be disabled by setting the number of retries to zero.
- **`Single-response`** – When ACOS is operating in parallel-query mode, ACOS sends two queries to the DNS server at the same time.

  Both queries could come back with valid responses. With this option enabled, the first valid response is forwarded to the client. If two invalid responses are received, the last one is forwarded to the client. By default, this option is enabled. When this option is disabled, if both responses are valid, ACOS forwards the responses to the server.
- **`Timeout`** – This option specifies the maximum number of seconds that ACOS waits for an AAAA response before sending an A query. You can specify 1-15 seconds, and the default is 1 second.
- **`Trans-ptr`** – This option helps you to run PTR queries for synthesized IPv6 addresses with the client.

  The PTR queries are intercepted by DNS64, converted into PTR queries for their corresponding IPv4 addresses, before being sent. When the response is received by ACOS, the response is synthesized and sent back to the client as if it were a response for the synthesized IPv6 address. By default, this option is disabled.
- **`TTL`** – This option specifies the maximum TTL to use in synthesized AAAA replies, instead of the TTL value in the original IPv4 DNS reply.

- If the TTL value in the template is lower than the TTL value in the IPv4 reply, the template's TTL value is used in the synthesized IPv6 reply.
- If the TTL value in the template is equal to or higher than the TTL value in the IPv4 reply, the TTL value in the IPv4 reply is used in the synthesized IPv6 reply.

By default, no TTL value is set in the template.

NAT64 and the Application Level Gateway

NAT64 has Application Level Gateway (ALG) support for the following protocols:

- Encapsulating Security Payload (ESP)
- File Transfer Protocol (FTP)
- H.323 standard (H323)
- Media Gateway Control Protocol (MGCP)
- Point-to-Point Tunneling Protocol (PPTP) Generic Routing Encapsulation (GRE)
- Real Time Streaming Protocol (RTSP)
- Session Initiation Protocol (SIP)
- Trivial File Transfer Protocol (TFTP)

ALG support for FTP is enabled by default. ALG support for the other protocols is disabled by default. However, Carrier Grade NAT (CGN) and Fixed NAT support these ALGs.

Fragmentation

Fragmentation is supported for packets that are larger than the Maximum Transmission Unit (MTU) of the inbound or outbound ACOS interface.

NOTE: Packet virtual reassembly is required for Carrier Grade NAT (CGN) devices to perform NAT and handle ALG traffic.

Generally, fragmentation for NAT64/DNS64 is useful in cases such as the following:

- The network has a lot of IPv6 fragments.
- Clients do not perform MTU path discovery, and are therefore forced to fragment packets that are too large.

The following fragmentation options are enabled by default:

- Inbound – Fragmentation of inbound IPv6 packets is enabled.
- Fragmentation of inbound IPv4 packets is disabled. If an inbound IPv4 packet has the Don't Fragment bit set, ACOS does not fragment the packet, and instead sends an ICMP unreachable message.
- Outbound – Fragmentation of outbound IPv4 packets is enabled.

Don't Fragment Bit

By default, NAT64 honors the Don't Fragment bit in inbound IPv4 packets. Optionally, you can configure NAT64 to override the Don't Fragment bit, and to fragment the packet anyway.

Fragment Interval and Queue Size

The maximum interval allowed between fragments is configurable. The maximum number of simultaneous fragmentation sessions ACOS will allow also is configurable.

DNS64 for 6rd Traffic

NAT64 and DNS64 support allows 6rd IPv6 clients to reach IPv4 servers.

No new configuration is required for the support. When ACOS receives a packet from a 6rd IPv6 client, ACOS checks whether the inner destination IPv6 address matches the VIP for DNS64, and the destination UDP port matches the virtual port on the DNS64 VIP.

Additional NAT Features

NAT64 also supports the following NAT features:

- Hairpinning
- User quotas
- Endpoint Independent Mapping (EIM)
- Endpoint Independent Filtering (EIF)
- Session Synchronization
- Logging

NAT64 shares configuration of these features with Large Scale NAT (LSN).

For more information, see the following:

- Large Scale Network Address Translation
- Traffic Logging Guide for IPv6 Migration

Configuring DNS64

Configuring DNS64 Using the GUI

1. To configure an IP Source NAT IPv4 pool, navigate to **ADC** > **IP Source NAT** > **IPv4 Pools**.
2. To configure an IP Source NAT IPv6 pool, navigate to **ADC** > **IP Source NAT** > **IPv6 Pools**.
3. To configure an IP Source NAT pool group, navigate to **ADC** > **IP Source NAT** > **Pool Groups.**
4. To configure a DNS template with DNS64 settings, navigate to **CGN** > **DNS64** > **Templates,** select **DNS**.
5. To add the configuration for the local DNS servers, navigate to **CGN** > **DNS64** > **Virtual Servers**, click **Create**.

Configuring DNS64 Using the CLI

Configure NAT Resources

1. To configure an IPv4 NAT pool, if DNS64 will be a proxy for a local IPv4 DNS server:

```
ACOS(config)# ip nat pool v4p 1.1.2.3 1.1.2.8 netmask /16
```

2. To configure an IPv6 NAT pool, if DNS64 will be a proxy for a local IPv6 DNS server:

```
ACOS(config)# ipv6 nat pool v6p 2010:db8::1 2010:db8::4a netmask 64
```

NOTE: Do not use the `lsn` option for the pools.

3. To configure an IPv6 ACL, (if both IPv4 and IPv6 local DNS servers will be proxied) enter the following commands. In this case, the ACL directs IPv6 traffic to the IPv6 pool instead of the IPv4 pool.

```
ACOS(config)# ipv6 access-list 2
ACOS(config-access-list:2)# permit tcp any any
```

For simplicity, the syntax for matching on all traffic is shown. You can use more restrictive matching if needed.

Configure NAT Settings

Scenario 1 - ACOS Device Provides DNS64 for IPv4 Local DNS Server

The following commands configure an IPv4 NAT pool, to enable the DNS VIP to reach the local IPv4 DNS server:

```
ACOS(config)# ip nat pool ipv4-pool1 10.1.1.100 10.1.1.100 netmask /24
```

Scenario 2 - ACOS Device Provides DNS64 for IPv6 Local DNS Server

The following command configures an IPv6 NAT pool and enable the DNS VIP to reach the local IPv6 DNS server:

```
ACOS(config)# ipv6 nat pool ipv6-pool1 4629::50 4629::50 netmask 64
```

Scenario 3 - ACOS Device Provides DNS64 for IPv6 and IPv4 Local DNS Servers

The following commands configure the DNS64 and NAT64 deployment shown in this figure.

NAT Configuration:

The following commands configure an IPv6 ACL that matches on all IPv6 traffic.

```
ACOS(config)# ipv6 access-list dnslist
ACOS(config-access-list:dnslist)# permit tcp any any
ACOS(config-access-list:dnslist)# permit udp any any
ACOS(config-access-list:dnslist)# permit icmp any any
ACOS(config-access-list:dnslist)# permit ipv6 any any
ACOS(config-access-list:dnslist)# exit
```

The following commands configure an IPv6 NAT pool and an IPv4 NAT pool.

NOTE: The IPv6 NAT pool enables the DNS VIP to reach the local IPv6 DNS server. The IPv4 NAT pool enables the DNS VIP to reach the local IPv4 DNS server.

```
ACOS(config)# ipv6 nat pool ipv6-pool1 4629::50 4629::50 netmask 64
ACOS(config)# ip nat pool ipv4-pool1 192.168.1.100 192.168.1.100 netmask
/24
```

Configure DNS Template

To configure a DNS template with DNS64 settings, enter the following commands:

```
ACOS(config)# cgnv6 template dns dns64-temp
ACOS(config-dns)# dns64 enable
```

This command creates the template and changes the CLI to the configuration level for the template.

The **dns64** command enables the DNS64 feature. For more information about the options, see DNS Template Options for DNS64.

Configure Server Settings

Scenario 1 - ACOS Device Provides DNS64 for IPv4 Local DNS Server

1. The following commands add a real server configuration for the IPv4 local DNS server:

```
ACOS(config)# cgnv6 server localdns-rs 10.20.32.10
ACOS(config-real server)# port 53 udp
ACOS(config-real server-node port)# exit
ACOS(config-real server)# exit
```

2. The following commands add the real server to a UDP service group.

```
ACOS(config)# cgnv6 service-group dns53 udp
ACOS(config-cgnv6 svc group)# member localdns-rs 53
ACOS(config-cgnv6 svc group-member:53)# exit
```

3. The following commands add the VIP that will receive DNS requests from IPv6 clients.

```
ACOS(config)# cgnv6 dns64-virtualserver vs1 3142::200
ACOS(config-cgnv6 vserver)# port 53 dns-udp
ACOS(config-cgnv6 vserver-vport)# source-nat pool ipv4-pool1
ACOS(config-cgnv6 vserver-vport)# service-group dns53
ACOS(config-cgnv6 vserver-vport)# template dns dns64-temp
```

Scenario 2 - ACOS Device Provides DNS64 for IPv6 Local DNS Server

1. The following commands add a real server configuration for the IPv6 local DNS server:

```
ACOS(config)# cgnv6 server localdns-rs 4629::1000
ACOS(config-real server)# port 53 udp
ACOS(config-real server-node port)# exit
ACOS(config-real server)# exit
```

2. The following commands add the real server to a UDP service group:

```
ACOS(config)# cgnv6 service-group dns53 udp
ACOS(config-cgnv6 svc group)# member localdns-rs 53
ACOS(config-cgnv6 svc group-member:53)# exit
ACOS(config-cgnv6 svc group)# exit
```

3. The following commands add the VIP that will receive DNS requests from IPv6 clients:

```
ACOS(config)# cgnv6 dns64-virtualserver vs1 3142::200
ACOS(config-cgnv6 dnsvserver)# port 53 dns-udp
ACOS(config-cgnv6 dnsvserver-vport)# source-nat pool ipv6-pool1
ACOS(config-cgnv6 dnsvserver-vport)# service-group dns53
ACOS(config-cgnv6 dnsvserver-vport)# template dns dns64-temp
```

Scenario3 - ACOS Device Provides DNS64 for IPv6 and IPv4 Local DNS Servers

The following commands configure the DNS64 and NAT64 deployment shown in this figure.

1. The following commands configure an IPv6 ACL that matches on all IPv6 traffic.

```
ACOS(config)# ipv6 access-list dnslist
ACOS(config-access-list:dnslist)# permit tcp any any
ACOS(config-access-list:dnslist)# permit udp any any
ACOS(config-access-list:dnslist)# permit icmp any any
ACOS(config-access-list:dnslist)# permit ipv6 any any
ACOS(config-access-list:dnslist)# exit
```

2. The following commands configure an IPv6 NAT pool and an IPv4 NAT pool.

NOTE: The IPv6 NAT pool enables the DNS VIP to reach the local IPv6 DNS server. The IPv4 NAT pool enables the DNS VIP to reach the local IPv4 DNS server.

```
ACOS(config)# ipv6 nat pool ipv6-pool1 4629::50 4629::50 netmask 64
ACOS(config)# ip nat pool ipv4-pool1 192.168.1.100 192.168.1.100
netmask /24
```

3. The following commands add a real server configuration for each local DNS server.

```
ACOS(config)# cgnv6 server localdns-rs1 4629::1000
ACOS(config-real server)# port 53 udp
ACOS(config-real server-node port)# exit
ACOS(config-real server)# exit
```

4. The following commands add the real servers to a UDP service group.

```
ACOS(config)# cgnv6 server localdns-rs2 10.20.32.10
ACOS(config-real server)# port 53 udp
ACOS(config-real server-node port)# exit
ACOS(config-real server)# exit
```

5. The following commands add the real servers to a UDP service group.

```
ACOS(config)# cgnv6 service-group dns53 udp
ACOS(config-cgnv6 svc group)# member localdns-rs1 53
ACOS(config-cgnv6 svc group)# member localdns-rs2 53
ACOS(config-cgnv6 svc group)# exit
```

6. The following commands add the VIP that will receive DNS requests from IPv6 clients.

```
ACOS(config)# cgnv6 dns64-virtualserver vs1 3142::200
ACOS(config-cgnv6 dnsvserver)# port 53 dns-udp
ACOS(config-cgnv6 dnsvserver-vport)# source-nat pool ipv4-pool1
ACOS(config-cgnv6 dnsvserver-vport)# service-group dns53
ACOS(config-cgnv6 dnsvserver-vport)# template dns dns64-temp
ACOS(config-cgnv6 dnsvserver-vport)# exit
ACOS(config-cgnv6 dnsvserver)# exit
```

Optionally, you also can configure DNS64 override policies for specific clients. (See Override DNS64 Settings for Specific Clients.)

Configuring NAT64

Configuring NAT64 by using the GUI

To configure NAT64 by using the GUI:

1. Configure a NAT pool (or group of pools) that contains the IPv4 address(es) to use for NATting traffic from IPv6 clients to IPv4 servers by completing the following tasks:
   - To create a LSN pool, navigate to **CGN** > **LSN** > **LSN Pools**.
   - To create a LSN pool group, navigate to **CGN** > **LSN** > **LSN Pool Groups**.

2. To configure a Limit ID (LID) and add the pool or pool group to the LID, navigate to **CGN** > **LSN** > **LSN-LID**.
3. To import or configure a class list that matches on IPv6 client addresses and map the addresses to the LID, navigate to **CGN** > **LSN** > **Class List**.
4. To configure the NAT64 prefix, navigate to **CGN** > **NAT64**.
5. To bind the class-list to the NAT64 feature, navigate to **CGN** > **NAT64**.
6. Navigate to **CGN** > **LSN** > **Interfaces** to enable one of the following options:
   - Inside NAT on the interface that is connected to the internal clients.
   - Outside NAT on the interface connected to the Internet.

Configuring NAT64 by using the CLI

Configuring the NAT64 Prefix Information

1. Enter the following command to apply a NAT64 prefix to a class-list:

```
ACOS(config)# cgnv6 nat64 prefix64:ff9b::/96 class-list 2
```

NOTE: This command configures a default class-list. If a NAT64 prefix is not bound to a specific class-list, it is implicitly bound to the default class-list.

Repeat this command for the NAT64 prefixes to be applied to the same class-list.

Multiple Prefixes

Example 1

This example shows multiple NAT64 prefixes associated with a single class list.

```
ACOS(config)# cgnv6 nat64 inside source class-list list1
ACOS(config)# cgnv6 nat64 prefix 2012:1::/96
ACOS(config)# cgnv6 nat64 prefix 2012:2::/96
ACOS(config)# cgnv6 nat64 prefix 2012:3::/96
```

Example 2

This example shows multiple NAT64 prefixes associated with different class lists.

```
ACOS(config)# cgnv6 nat64 prefix 2012:1::/96 class-list 1
ACOS(config)# cgnv6 nat64 prefix 2012:2::/96 class-list 2
ACOS(config)# cgnv6 nat64 prefix 2012:3::/96 class-list 3
```

Example 3

Example 3 is a mixed case: some NAT64 prefixes are bound to a single class list, and other prefixes are individually bound to different class lists.

```
ACOS(config)# cgnv6 nat64 inside source class-list list1
ACOS(config)# cgnv6 nat64 prefix 2012:1::/96
ACOS(config)# cgnv6 nat64 prefix 2012:2::/96
ACOS(config)# cgnv6 nat64 prefix 2012:3::/96
ACOS(config)# cgnv6 nat64 prefix 2012:4::/96 class-list list2
ACOS(config)# cgnv6 nat64 prefix 2012:5::/96 class-list list3
```

Configuring a NAT Pool (or group of pools)

An IPv4 source NAT pool that contains the IPv4 address(es) that are used for translating traffic from IPv6 clients to IPv4 servers.

Enter the following command to configure the IPv4 source NAT pool:

```
ACOS(config)# cgnv6 nat pool pool1 10.1.1.1 10.1.1.1 netmask /24
```

The maximum number of NAT pools that can be configured is 8000.

The maximum number of NAT IPs that can be configured in a single NAT pool is 4096 IPs.

Configuring a Limit ID (LID)

Enter the following command to configure the LID:

```
ACOS(config)# cgnv6 lsn-lid 1
```

Binding IPv4 NAT Pool to the LID

Enter the following command to bind the IPv4 NAT pool to the LID:

```
ACOS(config-lsn lid)# source-nat-pool pool1
```

Configuring a class-list

This step configures a class list that matches on IPv6 client addresses and maps them to the LID.

1. Enter the following command to configure the class list:

   ```
   ACOS(config)# class-list NAT64_CLIENTS
   ```

   If you enter a list name, ACOS will add the list to the running-config. If the list is large, you can enter a filename with the **file** option to save the list to a file. In this case, the list entires are not displayed in the running-config.

2. The following command add an entry to match on IPv6 client addresses:

   ```
   ACOS(config-class list)# ::/0 lsn-lid 1
   ```

   To match on all IPv6 addresses, specify the address with ::/0.

3. The class list will apply to packets from the inside NAT interface to the outside NAT interface. There can be at most 1 class list for this purpose.

Binding the class-list to the NAT64 Feature

The class list will apply to packets from the inside NAT interface to the outside NAT interface. There can be at most 1 class list for this purpose.

Enter the following command to bind the class list to the NAT64 feature:

```
ACOS(config-class-list)# cgnv6 nat64 inside source class-list NAT64_
CLIENTS
```

Enabling IPv6 Inside NAT on the Interface Connected to the IPv6 Clients

The following commands configure the IPv6 interface connected to the IPv6 clients:

```
ACOS(config)# interface ve 2
ACOS(config-if:ve2)# ipv6 address 2001::1/64
ACOS(config-if:ve2)# ipv6 nat inside
```

Enabling IPv4 Outside NAT on the Interface Connected to the IPv4 Internet

The following commands configure the IPv4 interface connected to the IPv4 Internet:

```
ACOS(config-if:ve2)# interface ve 3
ACOS(config-if:ve3)# ip address 10.1.1.254 /24
ACOS(config-if:ve3)# ip nat outside
```

Configuring IPv4 Identification Value for IPv6 to IPv4 Translation

ACOS NAT64 translation is enhanced to prevent translated IPv4 packets from being blocked during IPv4 identification checking on security devices. The following command is added to the CLI:

```
ACOS(config)# cgnv6 nat64 force-non-zero-ipv4-id all
```

This command enables a non-zero Identification field in the IPv4 packet header to be set if there is no IPv6 fragment header.

This applies for packet sizes greater than 88 bytes and less than or equal to 1280 bytes. The all option enables this behavior for packets of all sizes. This is disabled by default.

Configuring One-to-One NAT Support for NAT64

Using the GUI

To configure one-to-one NAT mappings by using the GUI:

1. To add a pool, navigate to **CGN** > **One-to-One NAT** > **Pools**.
   a. Click **Create**.
   b. Enter the Pool Name, Start Address, End Address, Netmask, and VRID.
   c. Select the **Shared** checkbox and indicate whether this pool belongs to the Shared Partition, Single Partition, or a Partition Group. If the pool belongs to a Single Partition, enter the partition name.
   d. Click **Create**.
   e. Repeat steps these steps to add two more pools called *pool_2* and *pool_3*.

2. To create a group, navigate to **CGN** > **One-to-One NAT** > **Pool Groups**.
   a. Click **Create**.
   b. Enter the Name of the group.
   c. Select the VRID from the drop-down list.
   d. Select the Group Member.
   e. Click **Create**.
3. To specify the mapping timeout, navigate to **CGN** >**One-to-One NAT** > **Global**.
4. To display information on the allocated, freed, and failed mappings, navigate to **CGN** > **One-to-One NAT** > **Stats**.
5. To display information on the active one-to-one NAT mappings that are dynamically created at runtime, navigate to **CGN** > **One-to-One NAT** > **Mappings**.
6. To display the statistics of one-to-one NAT pools, navigate to **CGN** > **One-to-One NAT** >**Pools**.

Using the CLI

CLI Output

The following is an example for configuring one-to-one NAT for NAT64:

1. To configure One-to-One for NAT64, enter the following commands:

```
ACOS(config)# class-list nat64
ACOS(config-class list)# 2001:db8::/64 lsn-lid 1
ACOS(config-class list)# exit
ACOS(config)# cgnv6 one-to-one pool p1 6.6.6.150 6.6.6.150 netmask /24
ACOS(config)# cgnv6 lsn-rule-list rule1
ACOS(config-lsn-rule-list)# ip 6.6.6.100/32
ACOS(config-lsn-rule-list-ip)# tcp port 80 action one-to-one-snat pool
p1
ACOS(config-lsn-rule-list-ip)# udp port 53 action one-to-one-snat pool
p1
ACOS(config-lsn-rule-list-ip)# icmp action one-to-one-snat pool p1
ACOS(config-lsn-rule-list-ip)# exit
```

```
ACOS(config-lsn-rule-list)# default
ACOS(config-lsn-rule-list-default)# tcp port 0 action set-dscp outbound
63
ACOS(config-lsn-rule-list-default)# udp port 0 action set-dscp outbound
63
ACOS(config-lsn-rule-list-default)# icmp action set-dscp outbound 63
ACOS(config)# cgnv6 nat64 prefix 64:ff9b::/96
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn-lid)# lsn-rule-list destination rule1
ACOS(config-lsn-lid)# exit
ACOS(config)# cgnv6 nat64 inside source class-list nat64
```

2. To change the one-to-one NAT mappings timeout value, enter the following commands:

```
ACOS(config)# cgnv6 one-to-one mapping-timeout20
```

Logging

You can enable logging for one-to-one NAT44 and NAT64 using the following command:

```
ACOS(config)#cgnv6 template logging log_template_name
ACOS(config-logging:log_template_nam)#log one-to-one-nat sessions
```

The log formats supported are CEF, ASCII, Compact, and RFC5424.

For enabling one-to-one NAT logging and viewing the log samples, see *Traffic Logging Guide*.

Additional Configuration Options

The following sections describe additional configuration options.

To configure them in the GUI, navigate to **CGN** > **NAT64**.

To configure them using the CLI, see the syntax information in these sections.

The following topics are covered:

Configuring Application Level Gateway Support

- Enter the following command to enable NAT64 ALG support for the ESP protocol:

```
ACOS(config)#  cgnv6 nat64 alg  esp  enable
```

- Enter the following command to enable NAT64 ALG support for ESP protocol and only these ESP traffic which have IKE traffic can pass through ACOS

```
ACOS(config)#  cgnv6 nat64 alg  esp  enable-with-ctrl
```

- Enter the following command to disable NAT 64 ALG support for the FTP protocol:

```
ACOS (config) # cgnv6 nat64 alg ftp disable
```

NAT64 ALG support for FTP protocol is enabled by default, and it supports the following additional options:

- trans-eprt-to-port Translate EPRT to PORT (default is enabled)
- trans-epsv-to-pasv Translate EPSV to PASV (default is enabled)
- trans-lprt-to-port Translate LPRT to PORT (default is enabled)
- trans-lpsv-to-pasv Translate LPSV to PASV (default is enabled)
- xlat-no-trans-pasv Skip PASV response translate XLAT (default is disabled)

NOTE: NAT64 ALG support for FTP and TFTP is enabled by default.

- Enter the following command to enable NAT64 ALG support for H.323 protocol:

```
ACOS(config)#  cgnv6 nat64 alg h323 enable
```

- Enter the following command to enable NAT64 ALG support for mgcp protocol:

```
ACOS(config)# cgnv6 nat64 alg mgcp enable
```

- Enter the following command to enable NAT64 ALG support for PPTP protocol:

```
ACOS(config)#  cgnv6 nat64 alg pptp enable
```

- Enter the following command to enable NAT64 ALG support for RTSP protocol:

```
ACOS(config)#  cgnv6 nat64 alg rtsp enable
```

- Enter the following command to enable NAT64 ALG support for SIP protocol:

```
ACOS(config)#  cgnv6 nat64 alg sip enable
```

- Enter the following command to enable NAT64 ALG support for TFTP protocol:

```
ACOS(config)#  cgnv6 nat64 alg tftp enable
```

SIP Support

SIP ALG is disabled by default. You can enable it separately for LSN, NAT64, DS-Lite, and Fixed NAT.

When SIP ALG support is enabled, ACOS creates full-cone sessions to establish NAT mappings for SIP clients, and performs the necessary IP address translations in the SIP packet headers. The full-cone sessions are created for the SIP Contact port and the Real-time Transport Protocol (RTP)/Real-time Control Protocol (RTCP) port.

STUN Timeout

For SIP Contact NAT mappings, the corresponding full-cone session's Session Traversal Utilities for NAT (STUN) timeout is set to the "Expires" value in the SIP Registration packet's payload.

For SIP RTP/RTCP NAT mappings, the corresponding full-cone session's STUN timeout is configurable. The RTP/RTSP STUN timeout can be 2-10 minutes. The default is 5 minutes.

Enter the following command to change the RTP/RTCP STUN timeout for full-cone sessions used for SIP NAT mappings:

```
ACOS(config)#  cgnv6 lsn alg sip rtp-stun-timeout  5
```

Configuring Fragmentation Options

By default, fragmentation of inbound IPv6 packets is enabled, and fragmentation of inbound IPv4 packets is disabled. If an inbound IPv4 packet has the **Don't Fragment** bit set, ACOS does not fragment the packet but sends an ICMP unreachable message instead. Fragmentation of outbound IPv4 packets is enabled.

To change NAT64 fragmentation settings, enter the commands described in this section.

Enabling or Disabling Fragmentation Support for Inbound Packets

Enter the following command to enable NAT64 fragmentation support for inbound packets to drop silently:

```
ACOS(config)#  cgnv6 nat64 fragmentation inbound  drop
```

The following list describes the options:

- The **df-set send-icmp** option enables sending of ICMP unreachable messages for inbound fragmented packets, and disallows overriding the Don't Fragment bit.
- The **drop** option drops inbound fragmented packets.
- The **ipv6** option enables fragmentation support for inbound IPv6 packets.
- The **df-set** option disallows override of the Don't Fragment bit.

By default, the **ipv6** and **df-set send-icmp** options are enabled.

Enabling or Disabling Fragmentation Support for Outbound Packets

Enter the following command to enable NAT64 fragmentation support for outbound packets to drop silently:

```
ACOS(config)# cgnv6 nat64 fragmentation outbound drop
```

The following list describes the options:

- The **drop** option drops outbound fragmented packets.
- The **ipv4** option enables fragmentation of outbound IPv4 packets.

- The **send-icmpv6** option enables sending of ICMPv6 unreachable messages for outbound IPv6 fragmented packets, and disallows overriding the Don't Fragment bit.

By default, the **ipv4** option is enabled.

Changing the Fragment Timeout

By default, DS-Lite allows up to 60000 milliseconds (ms) between receipt of each fragment of a fragmented packet.

Enter the following command to change the fragment timeout for IPv4:

```
ACOS(config)# ip frag timeout  100
```

Enter the following command to change the fragment timeout for IPv6:

```
ACOS(config)# ipv6 frag timeout  100
```

Changing the Fragment Session Capacity

By default, DS-Lite can queue up to 100,000 DS-Lite packet fragments.

Enter the following command to change the queue size:

```
ACOS(config)# ip frag max-reassembly-sessions 1000
```

You can specify the maximum number of simultaneous fragmentation sessions ACOS will allow. The specified maximum applies to both IPv4 and IPv6.

Providing NAT64 Special Fragment Handling

ACOS provides special fragment handling for NAT64.

- ACOS supports IPv6 packets that contain special fragment headers, and that have the more-fragments bit set to zero and the fragmentation-offset set to zero.
- In the IPv4-to-IPv6 direction, insertion of headers that have the more-fragments bit set to zero and the fragmentation-offset set to zero is disabled by default. You enable insertion of these headers for NAT64. In this case, the headers are inserted when the IPv4 Don't Fragment bit is not set.

Here are the commands:

- Enter the following command to enable insertion of headers that have the more-fragments bit set to zero and the fragmentation-offset set to zero:

```
ACOS(config)# cgnv6 nat64 fragmentation df-bit-transparency enable
```

This command is entered at the global configuration level of the CLI.

Configuring TCP Maximum Segment Size Clamping

NAT64 uses its own TCP maximum segment size (MSS) clamping. The TCP MSS specifies the maximum length, in bytes, of data that one SYN or SYN-ACK packet in a TCP connection can have. The MSS does not include the TCP or IP header.

MSS Clamping Methods

You can set TCP MSS clamping for NAT64 to be performed using one of the following methods:

- None – ACOS does not change the MSS value.
- Fixed value – ACOS changes the MSS to the specified length.
- Subtract – ACOS reduces the MSS if it is greater than the specified number of bytes.

  This option sets the MSS based on the following calculations (S - Value to subtract from the maximum MSS Clamping value and N - Minimum value of the MSS Clamping):

  - If MSS minus S is greater than N, subtract S from the MSS.
  - If MSS minus S is less than or equal to N, set the MSS to N.

    By default, the subtract method of MSS clamping is used with the following values:

    - S = 20 bytes
    - N = 476 bytes

      Using these values, the default MSS clamping calculations are as follows:

- If MSS minus 20 is greater than 476, subtract 20 from the MSS.
- If MSS minus 20 is less than or equal to 476, set the MSS to 476.

NOTE: TCP MSS clamping is supported with One-to-One NAT64.

Changing the MSS Clamping Method

Enter the following command to change the MSS clamping method for NAT64 to a fixed maximum value of 22:

```
ACOS(config)# cgnv6 nat64 tcp mss-clamp fixed 22
```

Disabling TCP Resets in Response to Invalid TCP Packets

By default, if ACOS receives an invalid TCP packet from the inside network, ACOS sends a TCP reset for the host session. Optionally, you can disable TCP resets from being sent in this situation.

Enter the following command to disable TCP resets in response to invalid TCP packet from the inside network:

```
ACOS(config)# cgnv6 nat64 tcp reset-on-error outbound disable
```

Configuring ICMP Unreachable Options

ACOS can send ICMP Unreachable messages in one of the following cases:

- A configured user quota is exceeded
- No NAT ports are available for mappings

By default, ACOS sends code type 3, code 13, administratively filtered, when a configured user quota is exceeded. Sending of ICMP Unreachable messages when no NAT ports are available for mappings is disabled by default.

Enter the following command to send ICMP on port unavailable with code type 3 and code 13, administratively filtered:

```
ACOS(config)# cgnv6 lsn icmp send-on-port-unavailable admin-filtered
```

Enter the following command to send ICMP on quota exceeded with code type 3 and code 13, administratively filtered:

```
ACOS(config)# cgnv6 lsn icmp send-on-user-quota-exceeded admin-filtered
```

The following list describes the options:

- The **admin-filtered** option configures ACOS to send an ICMP Unreachable message with ICMP code type 3, code 13, administratively filtered.
- The **host-unreachable** option configures ACOS to send an ICMP Unreachable message with code type 3, code 1 for IPv4, and type 1 code 3 for IPv6.
- The **enable** and **disable** options enable or disable sending of the messages.

Override DNS64 Settings for Specific Clients

You can override the DNS64 settings for specific clients, with the following override actions:

- Disable – Does not perform DNS64 processing on the client's DNS request. The client's request is forwarded to the DNS server, and the reply is sent to client without modification.
- Different prefix – Uses a different NAT64 prefix to synthesize IPv6 addresses in the reply to the client. You can use this option to load balance NAT64 service across multiple ACOS devices.
- Exclude answer – Drops AAAA replies that contain specific IPv6 addresses or prefixes. In this case, ACOS sends an A query on behalf of the client, then uses DNS64 to add synthesized IPv6 addresses in the reply before sending the reply to the client.

Configuring the Override DNS64 Settings

To configure any of these override actions:

1. Configure a class list that specifies the IPv6 addresses or prefixes on which to perform the override action.
   - For the disable or different prefix actions, the class list specifies IPv6 clients.
   - For the exclude answer action, the class list specifies the invalid server IPv6

addresses to disallow.

In the class-list entry, specify the GLID or LID that specifies the override action to apply to the matching addresses. (See the next step.)

2. Configure a policy template that refers to the class list.
3. Configure either a GLID, or a LID in a policy template, to specify the override action.
4. Bind the policy template to the DNS virtual port on the DNS server VIP.

The following sections describe the syntax for each step. For configuration examples, see Override of DNS64 Setting Examples.

NOTE: GUI configuration of DNS64 override actions is not supported in the current release.

Configuring the Class List

Enter the following command to configure a class list that specifies IPv6 addresses or prefixes on which to perform an override action:

```
ACOS(config)# class-list list1
```

Enter the following command to add an entry that maps matching IPv6 addresses to a LID:

```
ACOS(config-class list)# 2001:db8::/64 lsn-lid 2
```

Configuring the Policy Template

Enter the following command to configure the policy template:

```
ACOS(config)# cgnv6 template policy p1
```

This command changes the CLI to the configuration level for the policy template.

At this level, enter the following command to specify the class list in the policy template:

```
ACOS(config-policy)# class-list list1
```

Configuring the LID or GLID

Configuring a LID for NAT64 Override

Enter the following command to use a LID at this configuration level to specify the override action:

```
ACOS(config-policy-class-list:list1))# lid 1
ACOS(config-policy-class-list:list1-lid:1)# dns64 exclusive-answer
```

Configuring a GLID for NAT64 Override

NOTE: If you plan to use a GLID to specify the override action, use this section.

Enter the following command to configure the GLID:

```
ACOS(config)# glid 22
```

At this level, enter the following command to specify the override action to drop AAAA replies that contain specific IPv6 addresses or prefixes:

```
ACOS(config-glid:22)# dns64 exclusive-answer
```

Binding the Policy Template to the DNS Virtual Port

Enter the following command to bind the policy template to the DNS VIP's virtual port:

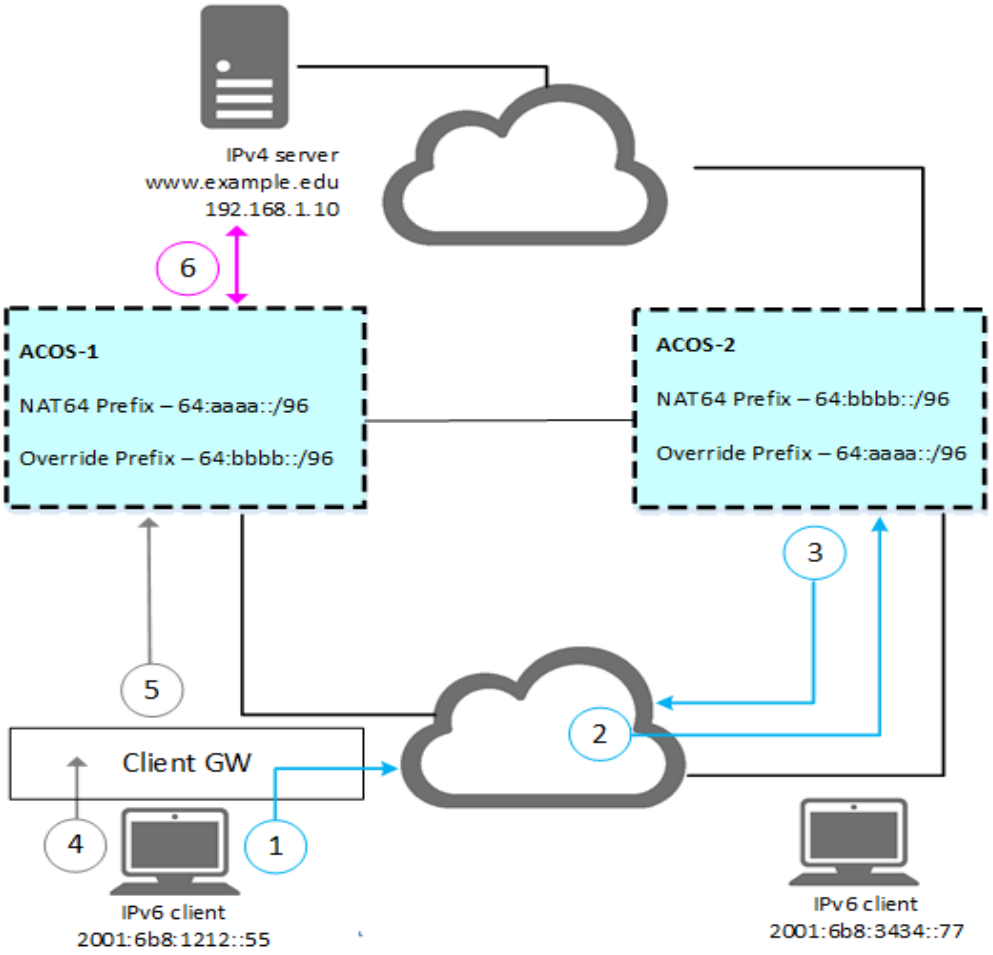
```
ACOS(config-cgnv6 dnsvserver-vport)# template policy p1
```

This command is entered at the configuration level for the virtual port.

NAT64 Load Balancing Using Override of the NAT64 Prefix

You can use the prefix override option to load balance NAT64 service among multiple ACOS devices, based on client IPv6 address. Figure 15 illustrates an example.

Figure 15 : NAT64 Load Balancing Using the Prefix Override Option



NOTE: This solution requires routes on the client gateway to direct client requests to one ACOS device or the other, based on the synthesized IPv6 server address to which the client sends the requests.

In this example, each ACOS device provides NAT64 for different NAT64 prefixes:

- ACOS-1 provides NAT64 for client requests to synthesized IPv6 server addresses with NAT64 prefix 64:**aaaa**::/96.
- ACOS-2 provides NAT64 for client requests to synthesized IPv6 server addresses with NAT64 prefix 64:**bbbb**::/96.

The client's initial DNS request can go to either ACOS device. The ACOS device that receives the initial client request checks the class list in the policy template bound to the DNS virtual port to see whether the client's IP address matches the class list.

One of the following actions is taken:

- If the client does not match the class list, the configured DNS64 prefix is used.
- If the client matches the class list, the override DNS64 prefix in the GLID is used instead.

When the client sends a request to the synthesized IPv6 server address, the route table on the client's gateway routes the request to a specific ACOS device, either ACOS-1 or ACOS-2, based on the synthesized IPv6 address. The ACOS device to which the client gateway routes this request is the ACOS device that will provide NAT64 for the client, enabling it to reach the IPv4 server.

The following procedure describes how NAT64 load balancing is performed for a specific client request (Figure 15):

1. IPv6 client 2001:6b8::55 sends a DNS request for the IPv6 address of site www.example.edu.

   The site resides on an IPv4 server.

2. Request arrives at the carrier network, and is sent to one of the ACOS devices.

   The initial request can be routed to either ACOS device. In this case, the request is sent to ACOS-2.

3. Client's IP address matches the class list that is bound to the DNS virtual port.

   Therefore, ACOS-2 uses the prefix in the GLID that is mapped to the client's IP address in the class list, to synthesize IPv6 addresses to replace the IPv4 addresses in the reply to the client's DNS request. The synthesized address is 64:aaaa::c0a8:10a.

4. ACOS-2 sends the modified reply to the client.
5. Client sends request to 64:aaaa::c0a8:10a.
6. Client's gateway routes the request to ACOS-1, based on the destination IPv6 prefix.

7. ACOS-1 provides NAT64 service for the client, which enables the client to communicate with the IPv4 server.

Commands on ACOS-1

The following command configures the NAT64 prefix:

```
ACOS(config)# cgnv6 nat64 prefix 64:aaaa::/96
```

NOTE: The NAT64 prefix is different on each ACOS device. For brevity, the rest of the standard NAT64 / DNS64 configuration is not shown.

The following commands configure the class list:

```
ACOS(config)# class-list exclusive
ACOS(config-class list)# 2001:6b8:1212::/32 glid 2
ACOS(config-class list)# exit
```

The following commands configure the GLID:

```
ACOS(config)# glid 2
ACOS(config-glid:2)# dns64 prefix 64:bbbb::/96
ACOS(config-glid:2)# exit
```

The following commands configure the policy template:

```
ACOS(config)# cgnv6 template policy prefix
ACOS(config-policy)# class-list exclusive
ACOS(config-policy)# exit
```

The following commands bind the policy template to the DNS virtual port on the DNS server VIP:

```
ACOS(config)# cgnv6 dns64-virtualserver dns1 3142::cafe:6
ACOS(config-cgnv6 dnsvserver)# port 53 dns-udp
ACOS(config-cgnv6 dnsvserver-vport)# template policy prefix
```

Commands on ACOS-2

```
ACOS(config)# cgnv6 nat64 prefix 64:bbbb::/96
ACOS(config)# class-list exclusive
ACOS(config-class list)# 2001:6b8:3434::/32 glid 2
ACOS(config-class list)# exit
ACOS(config)# glid 2
```

```
ACOS(config-glid:2)# dns64 prefix 64:aaaa::/96
ACOS(config-glid:2)# exit
ACOS(config)# cgnv6 template policy prefix
ACOS(config-policy)# class-list exclusive
ACOS(config-policy)# exit
ACOS(config)# cgnv6 dns64-virtualserver dns1 3142::cafe:6
ACOS(config-cgnv6 dnsvserver)# port 53 dns-udp
ACOS(config-cgnv6 dnsvserver-vport)# template policy prefix
```

Override of DNS64 Setting Examples

This section shows configuration examples for the following DNS64 override actions:

- Disable – See Disable DNS64 for Specific Clients.
- Exclude answer – See Exclude Specific IPv6 Server Addresses.
- Different prefix – See NAT64 Load Balancing Using Override of the NAT64 Prefix.

(For information about DNS64 override actions, see Override DNS64 Settings for Specific Clients.)

NOTE: For simplicity, these examples focus only on the configuration for the override options and do not include the DNS64 or NAT64 configuration.

Likewise, these examples use GLIDs to specify the override actions. If you prefer, you can specify the override actions in LIDs in the policy template instead.

Disable DNS64 for Specific Clients

The commands in this section disable DNS64 for IPv6 clients with prefix 3142::/64.

The following commands configure the class list:

```
ACOS(config)# class-list reject
ACOS(config-class list)# 3142::/64 glid 31
ACOS(config-class list)# exit
```

The following commands configure the GLID:

```
ACOS(config)# glid 31
ACOS(config-glid:31)# dns64 disable
ACOS(config-glid:31)# exit
```

The following commands configure the policy template:

```
ACOS(config)# cgnv6 template policy reject
ACOS(config-policy)# class-list reject
ACOS(config-policy)# exit
```

The following commands bind the policy template to the DNS virtual port on the DNS server VIP:

```
ACOS(config)# cgnv6 dns64-virtualserver corporate1 3142::cafe:1
ACOS(config-cgnv6 dnsvserver)# port 53 dns-udp
ACOS(config-cgnv6 dnsvserver-vport)# template policy reject
```

Exclude Specific IPv6 Server Addresses

The commands in this section reject AAAA replies that have any IPv6 address in the ANSWER section with prefix 2001:470::/32.

The following commands configure the class list:

```
ACOS(config)# class-list exclusive
ACOS(config-class list)# 2001:470::/32 glid 1
ACOS(config-class list)# exit
```

The following commands configure the GLID:

```
ACOS(config)# glid 1
ACOS(config-glid:1)# dns64 exclusive-answer
ACOS(config-glid:1)# exit
```

The following commands configure the policy template:

```
ACOS(config)# cgnv6 template policy exclusive
ACOS(config-policy)# class-list exclusive
ACOS(config-policy-class-list:exclusive)# exit
```

The following commands bind the policy template to the DNS virtual port on the DNS server VIP:

```
ACOS(config)# cgnv6 dns64-virtualserver local1 3142::cafe:5
ACOS(config-cgnv6 vserver)# port 53 dns-udp
ACOS(config-cgnv6 vserver-vport)# template policy exclusive
```

Displaying and Clearing Information

- To display one-to-one NAT mappings (with or without filter), enter the following commands:

```
ACOS# show cgnv6 one-to-one mappings
```

The following is the output for this command:

```
ACOS(config)# show cgnv6 one-to-one mappings
>Inside IPv4 Address    Inside IPv6 Address     NAT Address     Sessions
    Age    Pool
------------------------------------------------------------------------
----------------------------
>-                      3ff7::85                11.1.1.130           0
    600    pool_1
>10.1.1.2               -                       11.1.1.129           0
    600    pool_1
Total One-to-One NAT Mappings: 2
```

- To display one-to-one NAT pool statistics, enter the following commands:

```
ACOS# show cgnv6 one-to-one pool statistics
```

- To display one-to-one NAT mappings for a specific IPv6 inside address, enter the following commands:

```
ACOS(config)# show cgnv6 one-to-one mappings inside-address-
ipv62001:300::40
```

The following text is the output:

Pool	Total Address	Used Address	Free Address
pool_1	255	155	100
pool_2	255	30	225

- To display one-to-one NAT statistics, enter the following commands:

```
ACOS# show cgnv6 one-to-one statistics
Total One-to-One Mapping Allocated: 23456
Total One-to-One Mapping Freed: 23000
```

```
One-to-One Mapping Allocation Failure: 100
```

- To clear one-to-one NAT statistics, enter the following commands:

```
ACOS# clear cgnv6 one-to-one statistics
```

- To clear one-to-one NAT mappings, enter the following commands:

```
ACOS# clear cgnv6 one-to-one mappings
```

- To clear the mappings, enter the following commands:

```
ACOS# clear cgnv6 one-to-one mappings inside-address-ipv6  2001:300::40
```

  This option filters the mappings that match the specified inside ipv6 address that will be cleared.

- To display one-to-one NAT mappings for a specific IPv6 inside address, enter the following commands:

```
ACOS# show cgnv6 one-to-one mappings inside-address-ipv62001:300::40
```

- Enter the following command to display configuration information for NAT64 ALG:

```
ACOS# show cgnv6 nat64 alg  espconfig
```

- Enter the following command to display NAT64 ALG statistics:

```
ACOS# show cgnv6 lsn alg espstatistics
```

  Statistics are shown for ALG sessions for LSN, NAT64, and DS-Lite, as applicable.

- Enter the following command to clear ALG statistics:

```
ACOS# clear cgnv6 lsn alg  esp statistics
```

- Enter the following command to display DNS64 statistics:

```
ACOS# show cgnv6 dns64 statistics
```

- Enter the following commands to display NAT64 information:

```
ACOS# show cgnv6 nat64 alg
ACOS# show cgnv6 nat64 conversion
ACOS# show cgnv6 nat64 full-cone-sessions
ACOS# show cgnv6 nat64 statistics
ACOS# show cgnv6 nat64 user-quota-sessions
```

For a detailed list of the sub-options available to each of these commands, see the *Command Line Reference*.

Dual-Stack Lite

This chapter describes Dual-Stack Lite (DS-Lite) and how to configure it.

Consider the following information:

- DS-Lite is based on RFC 6333, Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion.

  DS-Lite also uses CGN standards for the NAT component. (See Large Scale Network Address Translation.)

- For information about matching and traffic handling based on destination, see Destination Based NAT.
- For information about logging, see the Traffic Logging Guide for IPv6 Migration.

  Also described in the Logging Guide is Fixed-NAT. Fixed-NAT is a log optimization feature that allocates NAT ports for each client from a predetermined (“fixed”) set of ports on the NAT address. For information, see the Traffic Logging Guide for IPv6 Migration.

The following topics are covered:

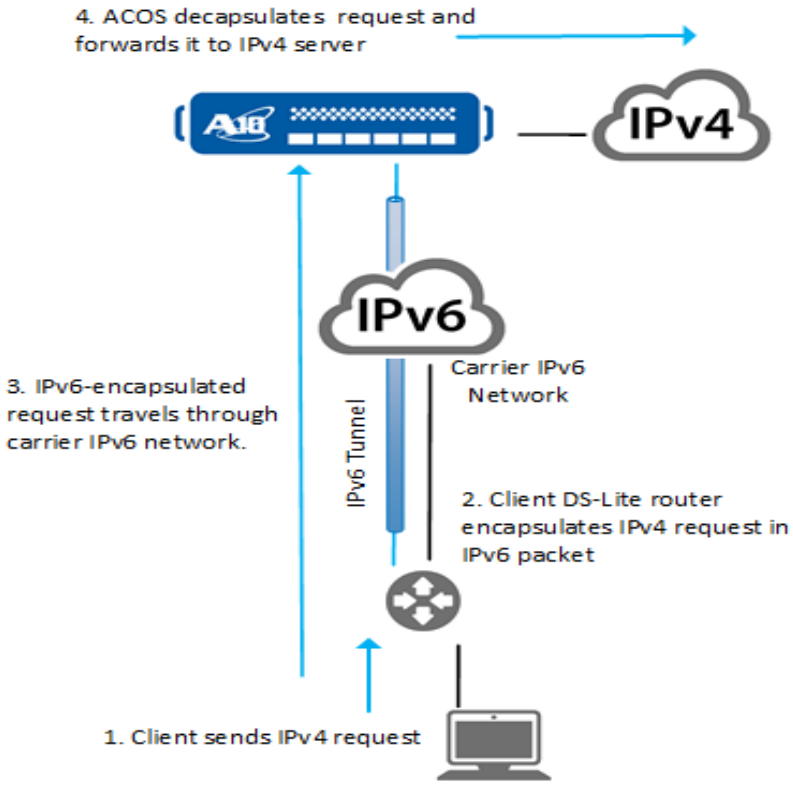
Overview

Dual-Stack Lite (DS-Lite) is a Network Address Translation (NAT) feature that enables the ACOS device to act as an end-point for IPv4 traffic tunneled through an IPv6 link.

Dual-Stack refers to the IP stacks for both IP versions, IPv4 and IPv6. Lite refers to the fact that this IPv4-IPv6 solution, which encapsulates IPv4 traffic in an IPv6 tunnel, is less complex than solutions that translate traffic between IPv4 and IPv6. DS-Lite can be used with Large Scale NAT (LSN) to provide NAT for large numbers of IPv4 clients that need NAT to reach IPv4 servers. Figure 16 shows an example of a DS-Lite deployment.

NOTE: Fixed NAT supports NAT44, NAT64, and DS-Lite.

Figure 16 : DS-Lite Example



In this deployment, an Internet carrier has an IPv6 network but uses DS-Lite to extend IPv4 as a service to its clients. Each client has a router that supports DS-Lite functionalities. Each client's DS-Lite router provides one end-point of the IPv6 tunnel through the carrier's network. The DS-Lite router encapsulates IPv4 traffic from the client in IPv6 packets and sends the IPv6 packets over the tunnel. The DS-Lite router decapsulates IPv4 traffic received over the tunnel before sending it to the client.

The ACOS device provides Address Family Transition Router (AFTR) functions for DS-Lite. The ACOS device decapsulates traffic exposing the client IPv4 address and translates the source IPv4 address using similar techniques as NAT44. The ACOS device encapsulates IPv4 traffic in IPv6 packets before sending the traffic over the tunnel to the client.

The tunnel endpoint on the ACOS device can be an Ethernet data interface loop-back address or a VRRP-A floating IP address.

Fragmentation Support

Fragmentation is allowed for packets that are larger than the Maximum Transmission Unit (MTU) of the inbound or outbound ACOS interface.

- By default, in the inbound direction (IPv4-IPv6 tunnel), IPv6 fragmentation is enabled. Fragmentation support for IPv4 packets in the IPv6 packets can be enabled in place of IPv6 fragmentation.
- By default, in the outbound direction, IPv4 fragmentation is enabled and cannot be disabled.

NOTE: Packet virtual reassembly is required for Carrier Grade NAT (CGN) devices to perform NAT and handle ALG traffic.

Don't Fragment Bit

By default, DS-Lite disregards the Don't Fragment bit in IPv4 packets that are destined for the IPv4 network, and in IPv6 tunnel packets. In either case, DS-Lite fragments the packet and does not send an ICMP unreachable message. You can also configure DS-Lite to send an ICMP unreachable message instead and to not fragment the packet.

Fragment Interval and Queue Size

The maximum interval allowed between fragments is configurable. The maximum number of simultaneous fragmentation sessions the ACOS device will allow also is configurable. (See Configuring Fragmentation Options.)

Application Level Gateway Support

DS-Lite provides Application Level Gateway (ALG) support for the following protocols:

- File Transfer Protocol (FTP)
- Trivial File Transfer Protocol (TFTP)
- Session Initiation Protocol (SIP)
- Real Time Streaming Protocol (RTSP)
- Point-to-Point Tunneling Protocol (PPTP) Generic Routing Encapsulation (GRE)

ALG support for FTP is enabled by default, and ALG support for the other protocols is disabled by default.

Consider the following information:

- If you are upgrading from legacy releases, ALG support for protocols other than FTP needs to be enabled explicitly in the configuration.
- When full-cone support is enabled for well-known ports, ALG support for TFTP still works even if TFTP ALG support is disabled.
- Session synchronization is not supported for ESP.

Configuring DS-Lite

You can configure DS-Lite by using the GUI or the CLI.

Configuring DS-Lite by Using the GUI

To configure DS-Lite, use the following GUI pages:

1. To configure NAT pools, navigate to **CGN** > **LSN** > **LSN Pools**
2. Optionally, to configure NAT pool groups, navigate to **CGN** > **LSN** > **LSN Pool Groups**
3. To configure Limit IDs (LIDs), navigate to **CGN** > **LSN** > **LSN LID**.

   For each LID, specify the NAT pool to use. Optionally, you can set user quotas for the LID.

4. To import or configure class lists for the user subnets that require DS-Lite, navigate to **CGN** > **LSN** > **Class Lists.**

   A class list is a list of internal subnets or hosts. In a class list, you can bind each internal subnet to an individual LID.

5. To bind a class-list to the DS-Lite feature, navigate to **CGN** > **LSN** > **Global** and select the Class List from the Class List Binding drop-down list.

   The class lists will apply to packets from the inside NAT interface to the outside NAT interface. There can be at most 1 class list used for this purpose.

6. To enable inside NAT on the interface connected (through the carrier's IPv6 network) to IPv4 clients, navigate to **CGN** > **LSN** >**Interface**.

   You can use the same menu path to enable outside NAT on the interface connected to the IPv4 Internet.

Configuring Additional Options Using the GUI

- To increase system resources, navigate to **System** > **Settings** > **Resource Usage**.
- To configure TCP maximum segment size clamping, fragmentation, or other options, navigate to **CGN** > **DS-Lite** > **DS-Lite Global**.

Configuring DS-Lite by Using the CLI

Configure a DS-Lite NAT Pools and Pool Groups

1. Enter the following command to configure a NAT pool:

```
ACOS(config)# cgnv6 nat pool dslite0 172.7.7.30 172.7.7.100 netmask /24
```

2. Enter the following command to configure a pool group:

```
ACOS(config)# cgnv6 nat pool-group group1
ACOS(config-pool-group)# member pool1
ACOS(config-pool-group)# member pool2
```

Configure Limit IDs (LIDs)

For each LID, you can specify the NAT pool to use and, optionally, set user quotas for the LID.

1. Enter the following command to configure a LID:

```
ACOS(config)# cgnv6 lsn-lid  11
```

2. Enter the following command to binds a DS-Lite (or LSN) NAT pool to the LID:

```
ACOS(config-lsn-lid)#  source-nat-pool  dslite0
```

3. Enter the following command to configure the IPv6 per-user mapping quota for each type of protocol supported for LSN (TCP, UDP, or ICMP):

```
ACOS(config-lsn-lid)#  user-quota  tco 100
```

   Use the **reserve** option to specify how many ports to reserve on a NAT IP for each user, if desired. If no value is specified, the reserve value is the same as the user-quota value.

NOTE: The user quote applies only to client IPv6 source addresses.

Configure the Class List for User Subnets that Require DS-Lite

A class list is a list of internal subnets or hosts. In a class list, you can bind each internal subnet to an individual LID.

Enter the following commands configure a class list to bind the client IPv6 addresses (the IPv6 addresses of the client DS-Lite routers) to the LID:

```
ACOS(config)# class-list dslite
ACOS(config-class list)# 2001::/16 lsn-lid 1
```

Bind a class-list for Use with DS-Lite

The class lists applies to packets from the inside NAT interface to the outside NAT interface. There can be at most 1 class-list for this purpose.

Binding the Class List for Use with DS-Lite

Enter the following command to bind the class list so that the list can be used with DS-List:

```
ACOS(config)# cgnv6 ds-lite inside source class-list dslite
```

Enable Inside NAT on the Interface Connected to IPv4 Clients

To enable the inside NAT on the interface that is connected (through the carrier's IPv6 network) to IPv4 clients, enter the following commands:

```
ACOS(config)# interface ethernet 1
ACOS(config-if:ethernet:1)# ipv6 nat inside
ACOS(config-if:ethernet:1)# exit
```

Enable Outside NAT on the Interface Connected to IPv4 Internet

To enable the outside NAT on the interface that is connected to the IPv4 Internet, enter the following commands:

```
ACOS(config)# interface ethernet 2
ACOS(config-if:ethernet:2)# ip nat outside
ACOS(config-if:ethernet:2)# exit
```

Configuring Filtering of Inside Client IPv4 Addresses Allowed to be NATed

By default, any inside IPv4 address that arrives in the IPv6 tunnel terminated by the ACOS device is allowed to be NATed. You can configure filtering of inside IPv4 addresses to specify the hosts or subnets that are permitted to be NATed. In this case, any IPv4 addresses that are not explicitly permitted by the filter are denied.

Configure a class list that contains the inside client IPv4 subnets or hosts that will be permitted to be NATed. Each entry must consist only of an IPv4 address and the mask length. In this example, the first entry permits any client in the 10.10.20.x /24 subnet. The second entry permits host 10.10.10.101. The following commands configure a class list to specify the inside IPv4 client addresses to allow to be NATed.

```
ACOS(config)# class-list client-permit
```

```
ACOS(config-class list)# 10.10.20.0/24
ACOS(config-class list)# 10.10.10.101/32
ACOS(config-class list)# exit
```

The following commands access the configuration level for the LSN LID used by DS-Lite, and enable client IPv4 filtering using the class list:

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn lid)# ds-lite inside-src-permit-list client-permit
```

Additional Configuration Options

The following topics are covered:

Increasing System Resources

ACOS allows you to adjust the system capacities for various resources, including the following resources that are critical to DS-Lite operation:

- Layer 4 sessions
- IP NAT pool addresses

The default maximum number allowed for these resources varies depending on your Thunder Series model (your ACOS device). To display the maximum for your ACOS device, use the **show system resource-usage** command.

Here is an example:

```
ACOS# show system resource-usage
Resource                          Current    Default    Minimum    Maximum
-------------------------------------------------------------------------
l4-session-count                  33554432   33554432   8388608
134217728
...
```

The **Current** column shows the maximum number currently allowed on the system. The default column shows the default maximum allowed.

NOTE: The maximum number of Layer 4 sessions for DS-Lite is actually half the displayed value. In this example, the current setting will support a theoretical maximum of 16,777,216 sessions. The maximum number of Layer 4 sessions includes user-quota sessions and full-cone sessions.

Enter the following command to change the maximum number of Layer 4 sessions that are allowed on the system:

```
ACOS(config)# system resource-usage l4-session-count  134217728
```

This command is entered at the global configuration level of the CLI.

The maximum value can be any value in the range between the values in the Minimum and Maximum columns in the **show system resource-usage** output.

NOTE: To place a system resource change into effect, reboot the ACOS device.

Configuring Static Port Mappings

Enter the following command to configure static mappings for a range of protocol ports for an IPv4 address:

The following command maps ports 80-100 on inside IP address 10.10.10.100 behind 2001:10::100 to 80-100 on NAT IP address 172.7.7.30. In this example, 2001:10::1 is the tunnel destination, which is the floating IP address on the ACOS device.

```
ACOS(config)# cgnv6 ds-lite port-reservation inside 2001:10::100
2001:db9::2:10 10.10.10.10 80 100 nat 192.168.210.45 80 100
```

The following command maps port 80 on the inside client to port 8080 on the NAT IP address:

```
ACOS(config)# cgnv6 ds-lite port-reservation inside 2001:10::8080
2001:db9::2:10 10.10.10.10 80 100 nat 192.168.210.45 80 100
```

Enabling Full-Cone Support for Well-Known Ports

By default, full-cone support is disabled for all ports. To enable full-cone support for these sessions, see Configuring Endpoint-Independent Filtering (EIF) and Mapping (EIM).

NOTE: Full-cone support is always provided for ports 1024-65535.

Configuring Application Level Gateway Support

Enter the following command to enable or disable ALG support for a protocol in DS-Lite:

```
ACOS(config)# cgnv6 ds-lite alg ftp disable
```

Configuring SIP Support

SIP ALG is disabled by default. You can enable it separately for LSN, NAT64, DS-Lite, and Fixed-NAT.

When SIP ALG support is enabled, the ACOS device creates full-cone sessions to establish NAT mappings for SIP clients, and performs the necessary IP address translations in the SIP packet headers. The full-cone sessions are created for the SIP Contact port and the Real-time Transport Protocol (RTP)/Real-time Control Protocol (RTCP) port.

Session lifetime and full-cone session lifetime are also snooped from registration packets to ensure sessions are not dropped while valid registration exists.

STUN Timeout

For SIP Contact NAT mappings, the corresponding full-cone session's Session Traversal Utilities for NAT (STUN) timeout is set to the "Expires" value in the SIP Registration packet's payload.

For SIP RTP/RTCP NAT mappings, the corresponding full-cone session's STUN timeout is configurable. The RTP/RTSP STUN timeout can be 2-10 minutes. The default is 5 minutes.

To change the RTP/RTCP STUN timeout for full-cone sessions used for SIP NAT mappings, use the following command at the global configuration level of the CLI:

```
ACOS(config)# cgnv6 lsn alg sip rtp-stun-timeout 5
```

Configuring Fragmentation Options

To change DS-Lite fragmentation settings, use the commands described in this section.

Enabling or Disabling Fragmentation Support

Enter the following commands to enable or disable DS-Lite fragmentation support:

```
ACOS(config)# cgnv6 ds-lite fragmentation inbound ipv6
ACOS(config)# cgnv6 ds-lite fragmentation outbound ipv4
```

The **inbound | outbound** option specifies the traffic direction. The **inbound** option applies to packets received on an ACOS interface. The **outbound** option applies to packets to be forwarded on an ACOS interface.

The **ipv4 | ipv6** option specifies the IP type.

Overriding the Don't Fragment Bit

Enter the following command to configure the DS-Lite response to packets that have the Don't Fragment bit set:

```
ACOS(config)# cgnv6 ds-lite fragmentation inbound df-set ipv4
ACOS(config)# cgnv6 ds-lite fragmentation outbound df-set ipv4
```

The **ipv4** option overrides the Don't Fragment bit for IPv4 packets destined for the IPv4 network. Likewise, the **ipv6** option overrides the Don't Fragment bit for IPv6

tunnel packets. With either option, DS-Lite does not send ICMP unreachable messages. Both the **ipv4** and **ipv6** options are enabled by default.

Changing the Fragment Timeout

By default, DS-Lite allows up to 60000 milliseconds (ms) between receipt of each fragment of a fragmented packet.

Enter the following command to change the fragment timeout:

```
ACOS(config)# ip  frag timeout  1000
ACOS(config)# ipv6  frag timeout  1000
```

Changing the Fragment Session Capacity

By default, DS-Lite can queue up to 100,000 DS-Lite packet fragments.

Enter the following command to change the queue size:

```
ACOS(config)# ip frag max-reassembly-sessions 4000
```

You can specify the maximum number of simultaneous fragmentation sessions the ACOS device will allow. The specified maximum applies to both IPv4 and IPv6.

Configuring TCP Maximum Segment Size Clamping

The TCP maximum segment size (MSS) specifies the maximum length, in bytes, of data that one SYN or SYN-ACK packet in a TCP connection can have. The MSS does not include the TCP or IP header.

Initially, the MSS is set by the IPv4 client in the SYN packet that the client sends to its DS-Lite router as part of the 3-way handshake to establish the TCP connection to a server. The MSS value that the client sets allows room for IPv4 and TCP headers. However, the IPv4 client typically does not also allow room for an IPv6 header.

On the ACOS device, DS-Lite must ensure that the server replies that are sent by the ACOS device onto the IPv6 tunnel to the client will have enough room for the data placed in the packet by the server. To verify the amount of room, DS-Lite checks the MSS value and, if necessary, changes it before sending the NATted request to the server. This process is called MSS clamping.

MSS Clamping Methods

You can set TCP MSS clamping for DS-Lite to be performed using one of the following methods:

- None – ACOS does not change the MSS value.
- Fixed value – ACOS changes the MSS to the specified length.
- Subtract – ACOS reduces the MSS if it is greater than the specified number of bytes.

This option sets the MSS based on the following calculations (S - Value to subtract from the maximum MSS Clamping value and N - Minimum value of the MSS Clamping):

- If MSS minus S is greater than N, subtract S from the MSS.
- If MSS minus S is less than or equal to N, set the MSS to N.

  By default, the subtract method of MSS clamping is used with the following values:

  - S = 40 bytes
  - N = 416 bytes

    Using these values, the default MSS clamping calculations are as follows:

    - If MSS minus 40 is greater than 416, subtract 40 from the MSS.
    - If MSS minus 40 is less than or equal to 416, set the MSS to 416.

Changing the MSS Clamping Method

Enter the following command to change the MSS clamping method for DS-Lite to a fixed maximum value of 10:

```
ACOS(config)# cgnv6 ds-lite tcp mss-clamp fixed 10
```

Disabling TCP Resets in Response to Invalid TCP Packets

By default, if the ACOS device receives an invalid TCP packet from the inside network, the ACOS device sends a TCP reset for the host session. An invalid TCP packet is

received without a matching session where a TCP RST will be sent. Optionally, you can disable TCP resets from being sent in this situation.

Enter the following command to disable TCP resets in response to invalid TCP packet from the inside network:

```
ACOS(config)# cgnv6 ds-lite tcp reset-on-error outbound disable
```

Configuring ICMP Unreachable Options

ACOS can send ICMP Unreachable messages in the following cases:

- A configured user quota is exceeded
- No NAT ports are available for mappings

By default, the ACOS device sends code type 3, code 13, administratively filtered when a configured user quota is exceeded. Sending of ICMP Unreachable messages when no NAT ports are available for mappings is disabled by default.

Enter the following command to change the behavior for either condition:

```
ACOS(config)# cgnv6 lsn icmp  send-on-port-unavailable  disable
```

Configuring Checksum Error Handling for Tunneled DS-Lite Traffic

As part of handling DS-Lite traffic, the ACOS device verifies the IP and Layer 4 checksums for IP packets encapsulated in the DS-Lite tunnel. You can specify the ACOS behavior when it detects an invalid IP or Layer 4 checksum in DS-Lite tunneled IP traffic.

For each type of checksum, you can specify one of the following behaviors:

- Fix – ACOS fixes the checksum and forwards the traffic.
- Drop – ACOS drops the traffic.
- Propagate (Layer 4 checksums only) – ACOS forwards the traffic without fixing the invalid checksum.

Layer 4 checksum handling applies to TCP, UDP, and ICMP packets encapsulated in a DS-Lite tunnel. Likewise, IP checksum handling applies to IPv4 packets encapsulated in a DS-Lite tunnel.

The default handling for IP checksum errors is to drop the packets. For Layer checksum errors, the default action is to propagate the packet.

Configuring Checksum Error Handling for Tunneled DS-Lite Traffic

NOTE: These options apply only to IP traffic that is encapsulated inside a DS-Lite tunnel. ACOS always drops other IPv4 traffic that has an invalid checksum.

- Enter the following command to configure ACOS handling of tunneled IPv4 traffic that has an invalid IPv4 checksum:

```
ACOS(config)# cgnv6 ds-lite ip-checksum-error  fix
```

  This command is entered at the global configuration level of the CLI.

- Enter the following command to configure ACOS handling of tunneled IPv4 traffic that has an invalid TCP, UDP, or ICMP Layer 4 checksum:

```
ACOS(config)# cgnv6 ds-lite l4-checksum-error  propagate
```

Configuring Zero UDP Checksum Handling

In certain scenarios like high-speed networks and real-time applications that use UDP as a tunnel encapsulation, the zero-checksum mode is enabled (See RFC 8085). To reduce the overhead of checksum calculation, you can configure ACOS to forward IPv4 data packets having zero UDP checksum without recalculating the checksum.

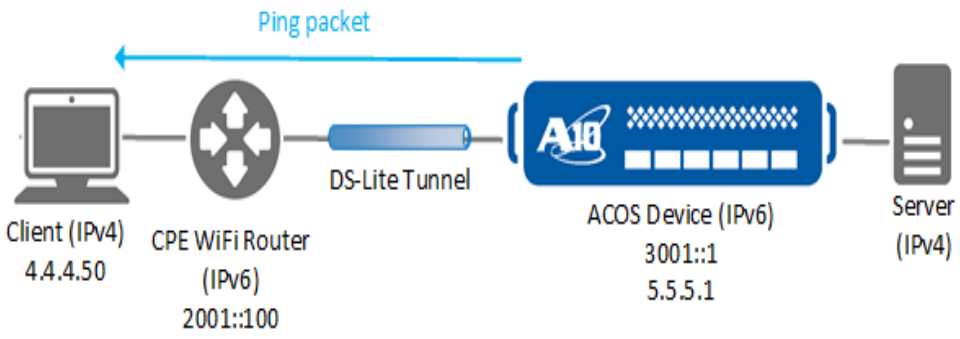
Enter the following command to forward IPv4 UDP packets with zero checksum without recalculation:

```
ACOS(config)# system udp skip-checksum-when-zero
```

Pinging a DS-Lite Client

ACOS allows the ACOS device to ping a client that is located behind Customer Premises Equipment (CPE) over a DS-Lite Tunnel. Figure 17 shows an example.

Figure 17 : Sending 'ping' to client over DS-Lite tunnel



The blue arrow in Figure 17 shows the ACOS device sending a ping packet over a DS-Lite tunnel, through the CPE equipment at the end-user's home, to the client at the other end.

Pinging a DS-Lite Client by Using the CLI

Enter the following command to ping a DS-Lite client:

```
ACOS# ping repeat 10 ds-lite source-ipv6 3001::1 source-ipv4 5.5.5.1
2001::100 4.4.4.50
```

This example relates to Figure 17. The following command is used to send 10 ping packets over a DS-Lite tunnel. The packets will have an IPv6 source address of 3001::1 and an IPv4 source address of 5.5.5.1. The remote client has an IP of 4.4.4.50.

You can also specify the IPv4 or IPv6 source address from which the ping packet(s) are generated. By default, the source IP will be the Ethernet interface from which the ping is sent.

- *remote-tunnel-addr* is the tunnel's endpoint. Specify an IPv6 address or a hostname.
- *remote-ipv4-addr* is the remote client's IP address. Specify an IPv4 address or a hostname for the client.

NOTE: Additional ping options, such as **flood**, **data**, and **repeat** are supported, but they must be specified in the CLI syntax before the **ds-lite** keyword.

Displaying and Clearing DS-Lite Information

- Enter the following commands to display configured class lists:

```
ACOS# show class-list  dslite
```

- Enter the following command to display Layer 4 port reservations:

```
ACOS# show cgnv6 ds-lite port-reservations
```

- Enter the following command to display currently active full-cone sessions:

```
ACOS# show cgnv6 ds-lite full-cone-sessions
```

- Enter the following command to display currently active user quota sessions:

```
ACOS# show cgnv6 ds-lite user-quota-sessions
```

- Enter the following command to display global statistics that are related to DS-Lite:

```
ACOS# show cgnv6 ds-lite statistics
```

- Enter the following command to clear DS-Lite statistics:

```
ACOS# clear cgnv6 ds-lite statistics
```

Port Batching

This chapter describes port batching and explains how to configure port batching v1 and v2.

The following topics are covered:

Overview

Port Batching is an option to reduce the volume of external traffic logs for IPv6 migration features. By allocating a set of multiple ports to the client during session initiation, Port Batching reduces the amount of data created by the ACOS device's logging features. Only a single log message is generated for the batch of ports.

Each time LSN allocates a port mapping for a client, a log message is generated. Port batching reduces logging by allocating a set of multiple ports to the client at the same time, and generating one log message for the batch of ports. When a port batch is assigned, a log message is generated. Similarly, when a port batch is freed, another log message is generated.

The following rules apply to port batching assignment:

- If a subscriber’s connections are fewer than the number of ports in a batch, then only one port batch is assigned.
- A new port batch is assigned only if all ports in the allocated port batches are depleted.
- A port batch can be freed only if all ports in a batch are freed.

Differentiating Port Batching v1 and v2

This section describes the differences between Port Batching v1 and v2.

1. Contiguous Port Batch Assignment in Port Batching v2

   In Port Batching v1, the ports in a batch are separated by a constant interval, for example, 1024, 1029, 1034, 1039. Depending on the data CPU size, the interval is different on different platforms.

   In Port Batching v2, contiguous port batch assignment is supported, for example, 1024, 1025, 1026, 1027.

2. Enhanced Capabilities in Port Batching v2:

- Maximum port batch size is increased to 4096.
- Warning logs can be generated when the usage of one port batch has reached the configured threshold.
- Simultaneous allocation of the same TCP and UDP batches is supported.
- NAT port range is configurable for NAT pools.

3. When configured with Port Batching v2 within an IP NAT pool, ACOS uses less memory and has better traffic processing performance.
4. When configured with Port Batching v2, contiguous ports in a batch are more manageable and usable by external logging analyzers.

Prerequisites

- After upgrading to the current release, users currently using Port Batching v1 can continue to use v1. port-batch-v1 is enabled automatically from startup-config if port-batch-v1 configurations are detected.

  However, it is strongly recommended that users plan on migrating to v2 for better performance.

- If no pre-existing port-batch-v1 configurations are detected, this feature is disabled by default.

  New users must explicitly use the **`cgnv6 enable-port-batch-v1`** command to enable Port Batching v1 manually, prior to configuring any of the following:

- Port-batching size
- NAT Pool configurations

Port Batching v1

Port Batching v1 is disabled by default. When you enable Port Batching v1, you can specify the number of ports to allocate in each batch.

The following batch sizes are supported:

- 1
- 8
- 16
- 32
- 64
- 128
- 256
- 512

NOTE: Port Batching requires CPU resources and can increase CPU utilization; for example, you may experience significant delays if you allocate 1 port for very large NAT pools. Be sure to plan accordingly when you configure port batching.

The Port Batching option sets the wait time for TCP port reuse. The wait time specifies how many minutes the ACOS device waits after a TCP port allocated as part of Port Batching becomes free, before re-allocating that port to another user in a new port batch. You can set the wait time to 0-10 minutes. The default is 2. If you set the wait time to 0, ports can immediately be reused.

Configuring Port Batching v1

You can configure Port Batching by using the GUI or CLI.

Configuring Port Batching Using the GUI

To configure port batching:

1. Navigate to **CGN** > **LSN** > **Global**.
2. In **Port Batching Size**, enter the number of ports to allocate in each batch.
3. Click **Update**.

Configuring Port Batching v1 Using the CLI

If no pre-existing port-batch-v1 configurations are detected, this feature is disabled by default.

Prerequisite

Use the **`cgnv6 enable-port-batch-v1`** command to explicitly enable Port Batching v1 if there is no pre-exisitng port-batch-v1 configuration in your deployment.

To configure Port Batching v1, enter the following commands at the global configuration level:

```
ACOS(config)# cgnv6 enable-port-batch-v1
ACOS(config)# cgnv6 lsn port-batching size 512
```

Port Batching V2

Port batches can be created in NAT pools using Large Scale Nat (LSN). This allows ACOS to assign port batches contiguously and increases the maximum configurable port batch size. The port range can be configured for the NAT pool and then configure up to 4096 ports per port batch. If a subscriber's connections are fewer than the number of ports in a batch, then only one port batch will be assigned. The only exception is when ALG connections need two consecutive ports in a batch, but the subscriber does not have two consecutive ports in any given batch. In that case, a new port batch will be assigned to the subscriber.

NOTE: To change the port batch size, all of the current configuration must be deleted, and all existing sessions need to be cleared first.

To support contiguous port batch assignments, NAT port ranges will be configurable within a NAT pool. In both the cases of a port batch and of a NAT pool, a warning log will be generated when a configurable usage threshold is reached. A log is generated when a port batch is allocated, and another log is generated when the port batch is freed. In the case that a session creation fails, the port batch allocation message will be immediately followed by a port batch freed log.

NOTE: When port batching is configured within an IP NAT pool, ACOS uses less memory and has better traffic processing performance.

Configuring Port Batching v2 in NAT Pools

To configure Port Batching v2 in a NAT pool in the CLI, enter the following commands at the global configuration level:

```
ACOS(config)# cgnv6 nat pool lsn 198.51.100.1 198.51.100.254 netmask /24
port-batch-v2-size 64 usable-nat-ports 1024 2000
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn lid)# source-nat-pool lsn
ACOS(config-lsn-lid)# exit
ACOS(config)# class-list lsn
ACOS(config-class list)# 5.5.5.0 lsn-lid 1
ACOS(config)# cgnv6 lsn inside source class-list lsn
```

NOTE: You must disable port overloading with the Port Batching v1 before using Port Batching v2 or Simple NAT Pool Port Overloading using the following command:

```
ACOS# no cgnv6 enable-port-batch-v1
```

To display logging information for IP NAT pool port batching, enter one of the following show commands:

```
ACOS# show cgnv6 logging keywords lsn port-batch-v2-allocated
ACOS# show cgnv6 logging keywords lsn port-batch-v2-freed
```

Simultaneous TCP/UDP Port Batch Allocation

In Port Batch version 2, TCP and UDP port batches with the same port range can now be assigned at the same time, even if only one protocol is used initially. For example, if a new user requires a TCP port, then a TCP port batch is allocated. The UDP port batch with the same port range will also be assigned to that user at that time. The TCP user quota is used to limit the port usage for inside users, and any configured UDP user quota is not applicable when this feature is enable. Additionally, NAT pools with TCP and UDP port batch allocation enabled cannot have an extended user quota configured as well.

NOTE:

- This feature is only supported in Port Batch version 2. The original Port Batching feature only assigns one protocol port batch at a time.
- Only a single log message will be generated when both the TCP and the UDP port batch are allocated together.

Configuring TCP/UDP Port Batch Allocation

This feature is configured at the IP NAT Pool configuration level. Port Batch version 2 must also be enabled for this feature to take effect. A new option, following in the configuration of Port Batch version 2, is added in the CLI. When configuring Port Batch version 2, enter the "**simultaneous-tcp-udp-batch-allocation**" option at the end of the command before committing the configuration to enable TCP and UDP port batches, like below:

```
ACOS(config)# cgnv6 nat pool lsn 198.51.100.1 198.51.100.254 netmask /24 port-batch-v2-size 64 simultaneous-batch-allocation
```

Configuration Example

The following configuration example configures an IP NAT pool named "portbatch2" and enables Port Batch v2, as well as simultaneous TCP and UDP port batch allocation. The IP NAT pool is added to an LSN LID 1. The LSN LID is then added to a class list called "portbatchlist", which is then applied to the IP NAT inside.

```
ACOS(config)# cgnv6 nat pool portbatch2 198.51.100.1 198.51.100.254
netmask /24 port-batch-v2-size 64 simultaneous-batch-allocation
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn lid)# source-nat-pool lsn
ACOS(config-lsn-lid)# exit
ACOS(config)# class-list portbatchlist
ACOS(config-class list)# 5.5.5.0 lsn-lid 1
ACOS(config)# cgnv6 lsn inside source class-list portbatchlist
```

Port Block Allocation Interim Logs

Port Batch version 2 logs are sent when a new port batch is allocated, and when the port batch is freed. In between the two log messages, you can choose to receive interim log messages. These are sent periodically based on a configurable time interval.

The interim log messages follow the same log format as the “port batch allocated” log. The only fields that change between interim logs are the uploaded and downloaded bytes field, and the duration for which the port batch is allocated to the subscriber. The uploaded and downloaded bytes display the aggregate amount of traffic that is served by the port batch since the port batch was first allocated. Since these numbers are aggregated, they do not display traffic information for each individual session within a port batch.

NOTE: If interim updates are enabled after a port batch has been created, then there will not be interim logs for that port batch. Interim logs will only be generated for port batches created after interim updates are enabled.

By default, the bytes and the duration of port allocation are not included in the logging messages. To include the port batch upload bytes, download bytes, and the duration in the logging messages, you must configure **`include-port-block-account`**in the logging template.

The port batch upload and download bytes are displayed in the Port Batch v2 Allocated and Freed messages. The duration of port batch allocated is displayed in the Port Batch v2 Interim-Update and Port Batch Freed messages.

For more information about including the upload and download bytes, and the duration of port allocation in the log messages, see *Traffic Logging Guide*.

Displaying Port Batching Statistics

1. Enter the following command to display port batch allocation statistics:

```
ACOS# show cgnv6 logging statistics
```

```
NAT Logging Statistics:
----------------------------
TCP Session Created                     0
TCP Session Deleted                     0
TCP Port Allocated                      0
TCP Port Freed                          0
TCP Port Batch Allocated                0
TCP Port Batch Freed                    0
UDP Session Created                     0
UDP Session Deleted                     0
UDP Port Allocated                      0
UDP Port Freed                          0
UDP Port Batch Allocated                0
UDP Port Batch Freed                    0
ICMP Session Created                    0
ICMP Session Deleted                    0
ICMP Resource Allocated                 0
ICMP Resource Freed                     0
ICMPV6 Session Created                  0
ICMPV6 Session Deleted                  0
ICMPV6 Resource Allocated               0
ICMPV6 Resource Freed                   0
--MORE--
```

Table 9 describes the fields in the command output.

Table 9 : Port batch allocation fields

Field	Description
TCP Port Batch Allocated	Number of TCP-port batches that have been allocated. Each allocation increments the counter by 1. For example, if the TCP batch size is 8, each batch of 8 that is allocated is counted as 1.
TCP Port Batch Freed	Number of TCP-port batches that have been freed.
UDP Port Batch Allocated	Number of UDP-port batches that have been allocated. Each allocation increments the counter by 1. For

Table 9 : Port batch allocation fields

Field	Description
	example, if the UDP batch size is 8, each batch of 8 that is allocated is counted as 1.
UDP Port Batch Freed	Number of UDP-port batches that have been freed.

2. In the output of the **show cgnv6 lsn user-quota-sessions** command, each allocated port is counted individually. For example, if a single batch of 8 TCP ports is allocated to the user, the count in the TCP column is 8. If a second batch of 8 ports is allocated to a user, the number of ports listed is 16.

```
ACOS# show cgnv6 lsn user-quota-sessions
LSN User-Quota Sessions:
Inside Address        NAT Address      ICMP    UDP     TCP     Session Pool
 LID  Flag
------------------------------------------------------------------------------
-------------------
8.8.8.8               15.15.15.15       0       0       1       1        p1
  2    -
Total User-Quota Sessions Shown: 1
```

3. The connection count (Conns column) in **show cgnv6 lsn full-cone-sessions** output shows the actual number of connections. For example, user 203.0.113.1:20001 has only one active TCP connection, even though the user w as allocated a batch of 8 ports. If the user quota is unusable, "U" is displayed under Flag.

```
ACOS#show cgnv6 lsn full-cone-sessions
LSN Full Cone Sessions:
Prot Inside Address    NAT Address           Outbnd Inbnd  Pool  CPU Age
   Flags
------------------------------------------------------------------------------
---------
TCP  8.8.8.8:50190     15.15.15.15:50190      1      0      p1    1   -
    -
Total Full-cone Sessions: 1
```

If PCP protocol is enabled, then “PCP” is displayed under Flag.

Limitation

Fixed NAT supports only Port Batching v1.

Protocol Port Overloading

The following topics are covered:

Overview

Port overloading allows one NAT mapping resource to be used by more than one flow when going to different destinations. The same NAT resource can be re-used as long as the configured destination is unique.

NOTE: A flow consists of an inside user IP address and a protocol port.

Port overloading is useful in cases where NAT resources are limited and the majority of the traffic is client-server traffic. The same NAT resources can be re-used for different sessions.

NOTE: NAT resource is defined as a combination of NAT IP and NAT port pairs.

When an outbound flow destination port is in the configured range of port overloading ports, ACOS allocates a NAT port from the NAT pool and marks this NAT port as capable of being overloaded by other flows. Port overloading starts when all ports are exhausted. Flow to ports which are not in the range of port overloading-enabled ports will be dropped if all ports are exhausted.

NOTE: If a NAT port is first used by a normal (not port-overloading) flow, then it cannot be used for port overloading later.

Port overloading is supported for LSN and GiFW.

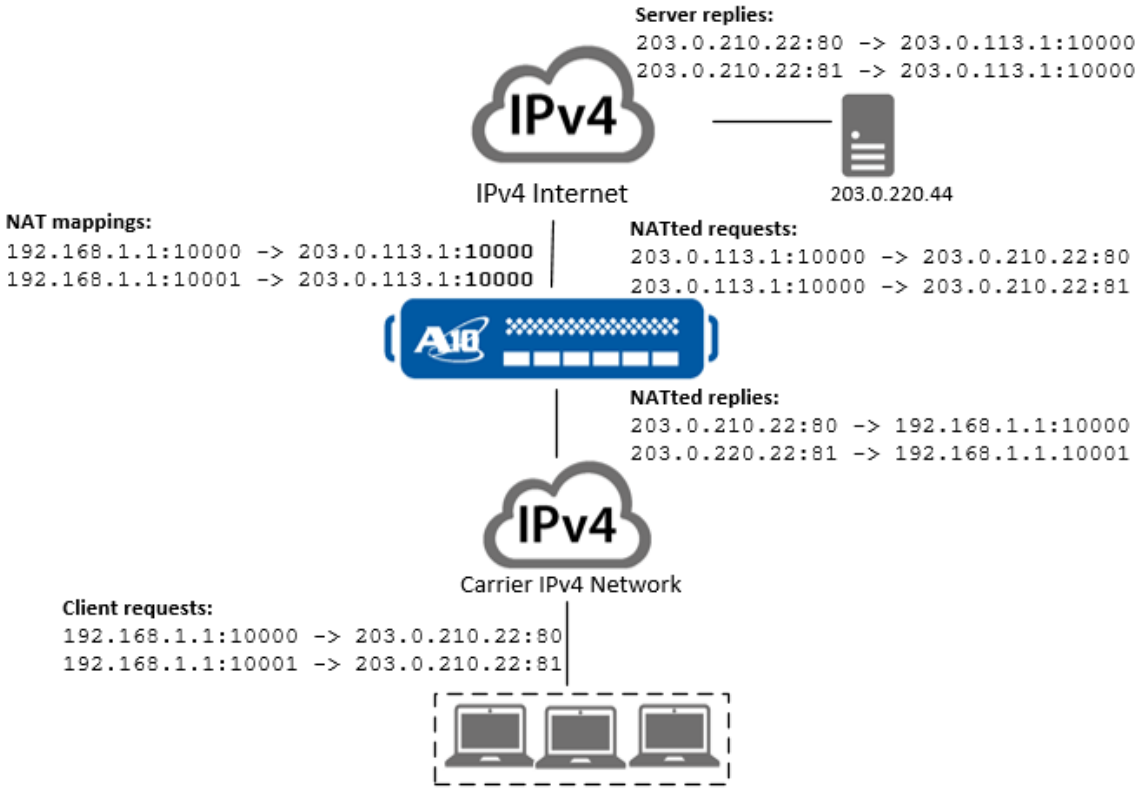
TCP and UDP Support

Only TCP and UDP protocols are supported on port overloading.

Unique Destination Address and Port

The same NAT mapping, which consists of a NAT IP address and protocol port, can be used for more than one destination, as long as the destination protocol port is unique. The destination IP address does not need to be unique. For example, the same mapping can be used for multiple connections to the same server.

Figure 18 : Port Overloading - IP address and port overloading



In this example, the granularity for port overloading is the IP address and the protocol port. A flow can use the same NAT IP address and NAT port when going to the same or a different destination IP address as long as the protocol port is unique. The client can use the same NAT IP address and NAT port for different flows that are being sent to the same server.

When the destination address and port 2-tuple are unique, CGN port overloading is in effect by default.

To illustrate this, the following is a sample configuration:

```
ACOS(config)# cgnv6 lsn port-overloading unique destination-address-and-
port
ACOS(config)# cgnv6 lsn port-overloading udp
ACOS(config-port-overloading-udp)# port 50000 to 60000
```

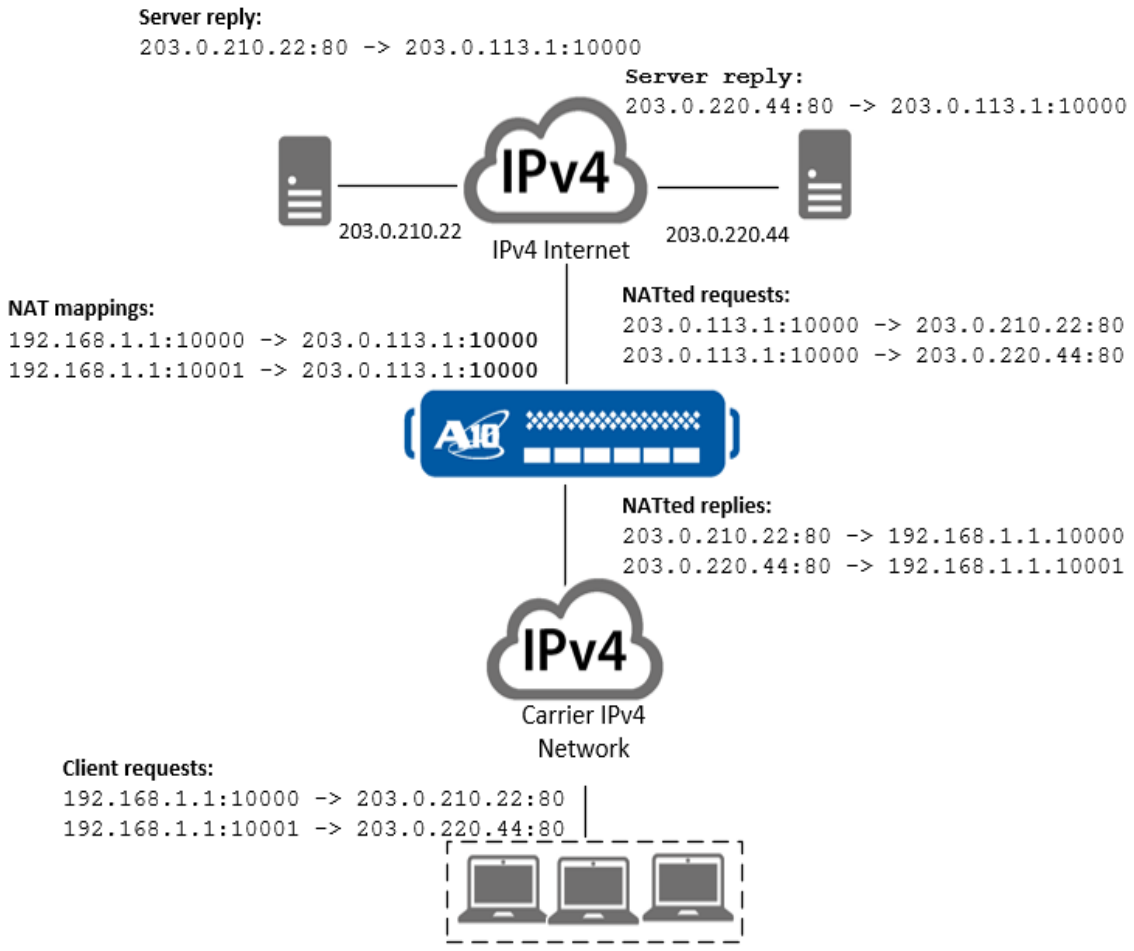
The following is an example of two sessions overloading to the same NAT resource, 60.1.12.12.1024 is re-used for different sessions:

```
Udp 61.1.1.107:20000 60.1.1.108:50000 60.1.1.108:50000 60.1.12.12:1024 300
1   NFe0f0r0 LSN
Udp 61.1.1.107:20000 60.1.1.108:50003 60.1.1.108:50003 60.1.12.12:1024 300
4   NFe0f0r0 LSN
```

Unique Destination Address

The same NAT mapping (NAT IP address and protocol port) can be used for more than one destination, as long as the destination IP addresses are unique. For example, the same mapping can be used for connection to two different servers but not for two connections to the same server.

Figure 19 : Port Overloading - IP address overloading only



In this example, the granularity for port overloading is the IP address only. The ACOS device can create more than one mapping for the client and use the same NAT IP address and protocol port for each mapping.

NOTE: The destination IP address must be unique for each mapping.

When the destination address is unique, CGN port overloading is in effect.

To illustrate this, the following is a sample configuration:

```
ACOS(config)# cgnv6 lsn port-overloading unique destination-address
ACOS(config)# cgnv6 lsn port-overloading udp
ACOS(config-port-overloading-udp)# port 50000 to 60000
```

The following is an example of two sessions overloading the same NAT resource, 60.1.12.12:1024 is reused for different sessions because 60.1.1.108 is different from 60.1.1.109. f the second session destination IP is 60.1.1.108, then port overloading is not allowed.

```
Udp 61.1.1.107:20000 60.1.1.108:50000 60.1.1.108:50000 60.1.12.12:1024 300
1  NFe0f0r0 LSN
Udp 61.1.1.107:20000 60.1.1.109:50003 60.1.1.109:50003 60.1.12.12:1024 300
4  NFe0f0r0 LSN
```

NOTE: To reduce redundancy in reload or reboot, you can use the `show cgnv6 lsn port-overloading config` command to verify the differences between the configured and actual settings.

Allow Different Users

By default, a port can be overloaded to create multiple mappings only for the same client. You can also enable ACOS to use the same overloaded port for more than one client.

In case of LSN, when all NAT resources are allocated to the existing users, a new user cannot access Internet as no NAT resource can be assigned to that user. Configuring **`allow-different-user`** enables the new user to use a NAT resource that is allocated to another user.

NOTE: The **allow-different-user** configuration is not applicable in case of Fixed NAT port overloading.
The **allow-different-user** option is not applicable for Port Batching v1 and v2 when the batch size is greater than 1.

By default, a port can be overloaded to create multiple mappings only for the same client. You can also enable ACOS to use the same overloaded port for more than one client. Use the **allow-different-user** command to allow a new user the access to NAT resource assigned to another user.

To illustrate this, the following is a sample configuration:

```
ACOS(config)# cgnv6 lsn port-overloading unique destination-address
ACOS(config)# cgnv6 lsn port-overloading allow-different-user
ACOS(config)# cgnv6 lsn port-overloading udp
ACOS(config-port-overloading-udp)# port 50000 to 60000
```

The following is an example of two sessions overloading the same NAT resource. 61.1.1.107 and 61.1.1.106 are different users, but they share the same NAT resource (60.1.12.12:1024):

```
Udp 61.1.1.107:20000 60.1.1.108:50000 60.1.1.108:50000 60.1.12.12:1024 300
 1  NFe0f0r0 LSN
Udp 61.1.1.106:20000 60.1.1.109:50003 60.1.1.109:50003 60.1.12.12:1024 300
 4  NFe0f0r0 LSN
```

EIM/EIF Considerations

Since full-cone sessions are created with the outside endpoints being unknown or changeable over time, the following considerations are important:

- A NAT port used for full-cone sessions cannot be used for port overloading.
- When a destination port resides within the range of ports enabled with port overloading, if this NAT port is used in ALGs (SIP/MGCP/h323) to create a full-cone session, then this port still cannot be used for port overloading.
- Since both port overloading and EIM/EIF are configured based on the destination port, the range used for port overloading cannot overlap with the range designated for EIM/EIF.

The following is a sample configuration:

```
ACOS(config)# cgnv6 lsn port-overloading udp
ACOS(config-port-overloading-udp)# port 1024 to 1200

ACOS(config)# cgnv6 lsn endpoint-independent-mapping udp
ACOS(config-eim-tcp)# port 1201 to 1300
```

ALG Control Session

Port overloading is only supported for ALG control session. Since the external endpoints of ALG data sessions are unknown, the destination IP needed to perform port overloading also becomes unknown. Therefore, porting overloading cannot be supported on ALG data sessions.

Limitations

- A NAT port used for full-cone sessions cannot be used for port overloading.
- The maximum number of times a port can be overloaded is limited to 127 for port batching v1, port batching v2, and simple NAT pools. For Fixed-NAT port batching, it is 65535.

Simple NAT Pool Port Overloading

A simple NAT Pool port supports Port Overloading for better traffic processing.

The following is a sample configuration for Simple NAT Pool port overloading:

```
ACOS(config)# cgnv6 lsn port-overloading unique destination-address
ACOS(config)# cgnv6 lsn port-overloading udp
ACOS(config-port-overloading-udp)# port 50000 to 60000
ACOS(config)# cgnv6 nat pool lsn 198.51.100.1 198.51.100.254 netmask /24
```

NOTE: Port Overloading supports a simple NAT pool structure even if the Port Batching v2 is not configured.

Fixed NAT Port Overloading

LSN and Fixed NAT share the same port overloading configuration and they share the same full-cone configuration for this purpose.

The following is a sample configuration for Fixed NAT port overloading:

```
ACOS(config)# cgnv6 lsn port-overloading udp
ACOS(config-port-overloading-udp)# port 1024 to 1200

ACOS(config)# cgnv6 lsn endpoint-independent-mapping udp
ACOS(config-eim-tcp)# port 1201 to 1300
ACOS(config)# cgnv6 lsn endpoint-independent-filtering udp
ACOS(config-eif-tcp)# port 1201 to 1300
```

In this example, Fixed NAT is enabled with port overloading on the UDP destination port 1024 to 1200 and the full-cone configuration is enabled on destination port 1021 to 1300.

Since NAT resources are pre-assigned to inside users, the **`allow-different-user`** option is not applicable to Fixed NAT. However, if a dynamic pool is configured for Fixed NAT, the NAT resources in the dynamic pool can be used and overloaded by other users.

In the following example, the following port mapping is used:

```
ACOS (config)# fixed-nat nat-address 2.1.1.1 port-mapping
NAT IP Address: 2.1.1.1
Inside User: 1.1.1.1
 TCP:  1024 to 32767
 UDP:  1024 to 32767
 ICMP: 1024 to 32767
Inside User: 1.1.1.2
 TCP:  32768 to 64511
 UDP:  32768 to 64511
 ICMP: 32768 to 64511
Dynamic Pool:
 TCP:  64512 to 65535
 UDP:  64512 to 65535
 ICMP: 64512 to 65535
```

The sample configuration is as follows:

```
ACOS(config)# cgnv6 fixed-nat inside 1.1.1.1 1.1.1.2 netmask /24 nat
2.1.1.1 2.1.1.1 netmask /24 dynamic-pool-size 1024
```

1. Dynamic port range 64512 to 65535 can be used and overloaded by both users, 1.1.1.1 and 1.1.1.2.
2. Port range 1024 to 32767 can only be used and overloaded by user 1.1.1.1.
3. Port range 32768 to 64511 can only be used and overloaded by user 1.1.1.2.

If all NAT ports assigned to an inside user is exhausted, the following occurs:

1. ACOS tries to allocate a NAT port from the dynamic pool, if one is configured.
2. If step 1 fails, port overloading is pushed to NAT ports assigned to that inside user.
3. If step 2 fails, port overloading is pushed to the dynamic pool, if one is configured.

Port Overloading and CGN Logging

When log port-overloading is configured under the CGN logging template level, ACOS logs port overloading port mapping events, namely PORT_ALLOCATED and PORT_ FREED.

For more information, see *Traffic Logging Guide for IPv6 Migration*.

Configuring Port Overloading

Port overloading is disabled by default. If you globally enable the feature, port overloading applies to all IPv6 migration features that use LSN NAT pools.

Consider the following information:

- Port overloading is not compatible with EIM or EIF.
- Before you use port overloading, ensure that EIM and EIF are disabled.
- To ensure that changes to the port overloading stage or to the granularity level take effect, you must load the software or reboot the device reboot.
- User-quota reserve values are applicable to port overloading.

- A new client can receive an allocation of ports only if at least the number of ports that are specified by the reserve value are available. If you plan to enable overloading of the same ports by multiple users, you must set the reserve value to 0.

Configuring Port Overloading by Using the GUI

1. Navigate to **CGN > LSN > Global**.
2. In Port Batching Size, enter the number of ports to allocate in each batch.
3. Configure the following options:
   - LSN Port Overloading
   - LSN Port Overloading Unique
   - LSN Port Overloading Allow Different User
4. Click **Update** to send the changes to the running-config.
5. Click **Save** to save the changes to the startup-config.

Configuring Port Overloading Using the CLI

1. Disabling the EIF and EIM using the following commands.

```
ACOS(config)# no cgnv6 lsn endpoint-independent-filtering tcp aa
ACOS(config)# no cgnv6 lsn endpoint-independent-mapping udp
```

2. Enabling the Port Batching v1 using the following command:

```
ACOS(config)# cgnv6 enable-port-batch-v1
```

3. Enabling the Port Overloading using the following command:

```
ACOS(config)# cgnv6 lsn port-overloading tcp
ACOS(config-port-overloading-tcp)# port 1 to 65535
```

Changing the Granularity

The default granularity is the destination IP address and protocol port.

1. Enter the following command to change the granularity to IP address only:

```
ACOS(config)# cgnv6 lsn port-overloading unique destination-address
```

The commands in this example implement the port overloading deployment in this figure.

2. Enter the following command to change the granularity to IP address and Protocol Port:

```
ACOS(config)# cgnv6 lsn port-overloading unique destination-address-and-port
```

The commands in this example implement the port overloading deployment shown in this figure.

Enabling Use of the Same Ports for Multiple Clients

By default, a port can be overloaded to create multiple mappings only for the same client.

Enter the following command to allow an overloaded port to be used by more than one client:

```
ACOS(config)# cgnv6 lsn port-overloading allow-different-user
```

NOTE: If you enable this option, port batching cannot be enabled. If Port Batching is enabled, you must disable it before you can enable the port overloading **allow-different-user** option.

You must set the user-quota reserve value in the LSN LID to 0. For example:

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn lid)# source-nat-pool p_lsn
ACOS(config-lsn lid)# user-quota udp 25 reserve 0
ACOS(config-lsn lid)# user-quota tcp 25 reserve 0
```

Displaying the Port Overloading Configuration

1. Enter the following commands to display the port overloading configuration changes that are ready to deploy, followed by the settings that are currently in effect:

```
ACOS(config)# show cgnv6 lsn port-overloading config
LSN Port-Overloading Configured:
cgnv6 lsn port-overloading unique destination-address
```

```
cgnv6 lsn port-overloading tcp disable well-known
cgnv6 lsn port-overloading tcp enable ephemeral
cgnv6 lsn port-overloading udp disable well-known
cgnv6 lsn port-overloading udp enable ephemeral
LSN Port-Overloading Actual:
cgnv6 lsn port-overloading disable
```

2. Enter the following commands to save the configuration changes to the startup-config, and reload the software to place the port overloading configuration changes into effect:

```
ACOS(config)# end
ACOS# write memory
Building configuration...
Write configuration to default startup-config
[OK]
ACOS# reload
Do you wish to proceed with reload? [yes/no]:yes
ACOS is reloading now. Please wait....
ACOS has reloaded successfully.
```

3. After the software reload is completed, enter the following command to verify that the port overloading configuration is now in effect:

```
ACOS >show cgnv6 lsn port-overloading config
LSN Port-Overloading Configured:
cgnv6 lsn port-overloading unique destination-address
cgnv6 lsn port-overloading tcp disable well-known
cgnv6 lsn port-overloading tcp enable ephemeral
cgnv6 lsn port-overloading udp disable well-known
cgnv6 lsn port-overloading udp enable ephemeral

LSN Port-Overloading Actual:
cgnv6 lsn port-overloading unique destination-address
cgnv6 lsn port-overloading tcp disable well-known
cgnv6 lsn port-overloading tcp enable ephemeral
cgnv6 lsn port-overloading udp disable well-known
cgnv6 lsn port-overloading udp enable ephemeral
```

NOTE:

- The configured settings do not take effect until the ACOS device is reloaded or rebooted.
- For Thunder 14045 devices, the output is displayed only for Master.
- For Thunder 7650 devices, the output is displayed only for one instance of the processing unit.

Port Control Protocol for LSN

This chapter describes how to configure Port Control Protocol (PCP) for LSN and on an ACOS device.

The following topics are covered:

Overview

When PCP is enabled, ACOS acts as a PCP server for LSN and PCP clients. The ACOS device parses incoming UDP packets that arrive on PCP port 5351, extracts the relevant information, and creates or refreshes the IPv4-IPv4 mapping as requested by the PCP client. The ACOS device sends a PCP response message to the PCP client.

NOTE: The mapping that is created for the client is an implicit dynamic mapping.

PCP DS-LITE Support for IPv6 Request

To support PCP requests for IPv6 packets from CPE, ACOS checks whether an IPv6 packet is sent from an NAT64 client or a DS-Lite Tunnel. ACOS tries to match the IPv6 source address with the DS-Lite inside class-list. If there is a match, ACOS then processes the request as a DS-Lite PCP request and extracts the IPv4 address from the PCP third-party option. If the third-party option exists and an IPv4 address is provided, ACOS assumes the PCP request is sent from CPE on behalf of the DS-Lite client.

ACOS then allocates the NAT IP/port and sends back the PCP response.

Configuration Options

You can use the following PCP options:

- Third-party

  You can enable this option by configuring an LSN PCP template, enabling the option in the template, and activating the template as the default PCP template.

- Prefer_failure
- Filter

Configuring Port Control Protocol

ACOS supports RFC 6887- compliant Port Control Protocol (draft 29). For more information, see RFC 6887.

NOTE: Draft versions 12 and 13 are no longer supported.

ACOS implements a PCP server for the following scenarios:

- Carrier-Grade NAT (NAT44)
- Dual-Stack Lite (DS-Lite)
- NAT64
- Fixed-NAT NAT44/DS-Lite/NAT64

You can create the following configurations on the PCP server:

- Enable or disable the process for MAP/PEER/ANNOUNCE Opcode. By default, the configuration is enabled.
- Set a minimum or maximum lifetime of mapping
- Allow a THIRD_PARTY request. By default, the configuration is disabled.
- Allow a THIRD_PARTY request coming from WAN interfaces. By default, the configuration is disabled.
- Enable or disable the validation of PCP MAP NONCE. By default, the configuration is disabled.
- Enable or disable the process to FILTER in PCP MAP. By default, the filter is enabled.
- Set the server listening UDP port. The default port is 5351.
- Allow the ACOS device to send an unsolicited announce packet when the ACOS device reboots or reloads or following VRRP-A failover. By default, the configuration is disabled.

PCP Requests

PCP packets are transported by using User Datagram Protocol (UDP). The port 5351 is normally used as PCP server listening port, while 5350 used as client listening port.

Sample Work flow for PCP Requests

1. ACOS accepts a PCP request from a LAN interface or, when ACOS is configured to permit a third-party PCP request that comes from a WAN, a WAN interface.
2. ACOS parses and validates this request

   ACOS silently drops the packet in the following situations:

   - The packet is not a PCP request.
   - The packet is too short to be a valid PCP request.
   - The packet is for a WAN interface, but ACOS is not configured to receive this packet.
   - There is no route to send a return response.
3. ACOS processes the request.

   If the request is successfully processed, for example a mapping for the MAP request is created, the server returns a success response. If an error occurs during the process, the server returns an error response.

PCP Result Codes

These are the PCP result codes that might be displayed by ACOS:

- 0 SUCCESS: Success.
- 1 UNSUPP_VERSION: The version number at the start of the PCP Request header is not recognized by this PCP server. This document describes PCP version 2.
- 2 NOT_AUTHORIZED: The requested operation is disabled for this PCP client, or the PCP client requested an operation that cannot be fulfilled by the PCP server's security policy, e.g., the client IP is not in LSN Class-List.
- 3 MALFORMED_REQUEST: The request could not be successfully parsed.
- 4 UNSUPP_OPCODE: Unsupported Opcode.
- 5 UNSUPP_OPTION: Unsupported option. This error only occurs if the option is in the mandatory-to-process range.
- 6 MALFORMED_OPTION: Malformed option.
- 7 NETWORK_FAILURE: The PCP server or the device it controls is experiencing a network failure of some sort
- 8 NO_RESOURCES: Request is well-formed and valid, but the server has insufficient resources to complete the requested operation at this time.

For example, the NAT pool is exhausted.

- 9 UNSUPP_PROTOCOL: Unsupported transport protocol, e.g., SCTP.
- 10 USER_EX_QUOTA: This attempt to create a new mapping would exceed this subscriber's port quota.
- For example, lsn user-quota is exceeded.
- 11 CANNOT_PROVIDE_EXTERNAL: The suggested external port and/or external address cannot be provided. This error MUST only be returned for MAP requests that included the PREFER_FAILURE option.

  The following situations can cause this error:

  - If there is a *prefer_failure* in the map request, and the request IP or port number cannot be allocated.
  - The request IP or port number cannot be allocated for the peer request.
- 12 ADDRESS_MISMATCH: The source IP address of the request packet does not match the contents of the PCP Client's IP Address field, due to an unexpected NAT on the path between the PCP client and the PCP-controlled NAT or firewall.
- 13 EXCESSIVE_REMOTE_PEERS: The PCP server was not able to create the filters in this request. This result code MUST only be returned if the MAP request contained the FILTER option.

  ACOS supports a maximum of 3 filters.

PCP MAP Opcodes

The MAP Opcode is used to create an explicit mapping between the following:

- An Internal Address and a port
- An External Address and a port

NOTE: On ACOS, mapping is also known as the LSN Full-cone session.

Sample Work flow for MAP Requests

1. The ACOS device checks whether the client is authorized based on configuration.

   For example, the client should live in one of the following:

- LSN/NAT64/DS-Lite Class-List
- Fixed NAT inside
- Static NAT

2. Depending on the requested mapping's lifetime value, one of the following actions is taken:
   - If the requested mapping's lifetime is not zero, this is a request to create or update a mapping.
   - If the requested lifetime is zero, this is a request to delete an existing mapping.
3. If a mapping exists for the requested internal address + protocol + port, ACOS completes the following tasks:
   - When "check-client-nonce" is enabled, ACOS first check the "nonce" value in request, if it does not match the nonce value of existed mapping, ACOS return NOT_AUTHORIZED.
   - If the MAP request contains PREFER_FAILURE option, but the suggested external address and port do not match those of existed mapping, ACOS returns CANNOT_PROVIDE_EXTERNAL.

     If it is a PEER request, but the suggested external address and port does not match those of an existed mapping, ACOS returns CANNOT_PROVIDE_ EXTERNAL.
   - If the MAP request does not contain PERFER_FAILURE, the ACOS device returns success, an external address, and the port of existed mapping in its response.
   - If the existing mapping is static, for example, static NAT, ACOS returns success, an external address, the port of the static NAT in its response, and indicate a lifetime of 2^32-1 seconds.
   - ACOS updates the existing mapping's lifetime based on new request.
   - ACOS updates the existing mapping's FILTER based on new request.
4. If no mapping exists for the internal address, protocol, and port, and ACOS creates a mapping by using the suggested external address and port.

   If no mapping exists for the internal address, protocol, and port:

- When it is an MAP request, ACOS creates a mapping by using the suggested external address and port.
- When it is a PEER request, if the requested external IP address and port number are valid for this client, ACOS creates a mapping by using the suggested external address and port. If the requested external IP address and port number are not valid for this client, ACOS returns CANNOT_PROVIDE_ EXTERNAL.

  ACOS might not be able to create a new mapping by using the suggested external address and port in the following situations:

  - The suggested external address does not belong to ACOS device.
  - The suggested external address, protocol, and port are in use.
  - The suggested external address, protocol, and port is prohibited

    For example, the port is less than 1024.

    - The suggested external IP address, protocol, or suggested port are invalid or invalid combinations.

      For example, external address 127.0.0.1, ::1, a multicast address, or the suggested port is not valid for the protocol.

5. If the PCP server cannot assign the suggested external address, protocol, and port, the following actions occur:
   - If the request contains the PREFER_FAILURE option, ACOS returns CANNOT_ PROVIDE_EXTERNAL.
   - If the request does not contain the PREFER_FAILURE option, one of the following occurs:
     - For MAP requests, ACOS assigns another external address and port for that protocol and returns the newly assigned external address and port in the response.
     - For PEER requests, ACOS returns CANNOT_PROVIDE_EXTERNAL, because the PREFER_FAILURE option is automatically implied by PEER requests.

Options for MAP or PEER Opcodes

The following options are available for MAP or PEER Opcodes:

- THIRD_PARTY

  This option is used when a PCP client wants to control a mapping to an internal host other than itself. This is used with both MAP and PEER Opcodes.

- PREFER_FAILURE

  This option is only used with the MAP Opcode.

  This option indicates that if the PCP server is unable to map both the suggested external port and suggested external address, the PCP server should not create a mapping.

- FILTER

  This option is only used with the MAP Opcode.

  This option indicates that filtering incoming packets is desired.

  After processing this MAP request containing the FILTER option and generating a successful response, the PCP-controlled device will drop packets received on its public-facing interface that don't match the filter fields. After dropping the packet, if its security policy allows, the PCP-controlled device MAY also generate an ICMP error in response to the dropped packet.

Rapid Recovery

PCP clients can to repair failed mappings in seconds. Mapping failures might occur in one of the following scenarios:

- When a NAT gateway is rebooted and loses its mapping state.
- When a NAT gateway has its external IP address changed so that its current mapping state becomes invalid.

Rapid recovery can be completed in one of the following ways:

- ANNOUNCE Opcode

  When the PCP server loses its state (for example, when it rebooted), it resets its Epoch time to its initial starting value (usually zero) and sends an ANNOUNCE response to the link-scoped multicast address through the LAN interface by using the configured source-ip/ipv6 address.

After the PCP server receives the ANNOUNCE Opcode request from the client and successfully parses and processes it, the server generates a SUCCESS response. This process allows the PCP client to determine the server's running state.

- PCP Mapping Update

  This rapid recovery method is used when the PCP server determines its existing mapping are invalid.

  This method are useful for servers that are routinely reconfigured by an Administrator or have their WAN address changed frequently will implement this feature (e.g., residential CPE routers).

NOTE: ACOS devices do not need to use this method.

Determining the Mapping Lifetime

You can configure how long a mapping lasts.

When the PCP client requests a certain mapping lifetime, the PCP server grant a lifetime which may be smaller of larger than the requested lifetime. You can configure the minimum and maximum lifetime values on the PCP server. The minimum value is 120 seconds, and the maximum value is 24 hours.

NOTE: On ACOS devices, a mapping cannot be deleted when there is an active session that uses the mapping.

Configuring Port Control Protocol

You can configure PCP using the following procedures.

Configuring Port Control Protocol by Using the GUI

1. Define a PCP template and set it as a the default PCP template:
   a. Navigate to **CGN > LSN > Templates > PCP**.
   b. Click **Create**.

c. Enter the **Name** of the PCP Template.
d. Enter or select options, as desired.
e. When finished, click **Create**.
f. Navigate to **CGN > LSN > Global**.
g. Select the PCP template from the Default PCP Template drop-down list.

2. Click **Save** to save the changes to the startup-config.

Configuring Port Control Protocol by Using the CLI

To complete each of the following tasks, enter the appropriate command:

- To define a PCP template to reuse a set of PCP options:

```
ACOS(config)# cgnv6 template pcp11
```

At the configuration level for the template, use the following commands to configure options.

- To apply options to set the listening UDP port for PCP packets:

```
ACOS(config-pcp:11)# pcp-server-port5351
```

- To enable or disable support for MAP/PEER/ANNOUNCE Opcode:

```
ACOS(config-pcp:11)# disable-opcodeannounce
```

- To set the mapping minimum and maximum lifetime values in minutes:

```
ACOS(config-pcp:11)# mapping-lifetimeminimum 5
ACOS(config-pcp:11)# mapping-lifetimemaximum 10
```

  - The configurable range of minimum/maximum mapping lifetime is 2-1440 minutes, and the minimum lifetime cannot be larger than maximum lifetime value.
  - If PCP client requested mapping lifetime is less than minimum lifetime value, ACOS uses the minimum lifetime as assigned mapping lifetime.
  - If PCP client requested mapping lifetime is larger than maximum lifetime value, ACOS uses the maximum lifetime as assigned mapping lifetime.

- To allow the THIRD_PARTY option:

```
ACOS(config-pcp:11)# allow-third-party-from-lan
ACOS(config-pcp:11)# allow-third-party-from-wan
```

By default, the **allow-third-party** options are disabled.

- To allow the ACOS device to send an unsolicited PCP ANNOUNCE message when the device is rebooted or following VRRP-A failover:

```
ACOS(config-pcp:11)# send-unsolicited-announce source-ip ipv4addr
ACOS(config-pcp:11)# send-unsolicited-announce source-ipv6 ipv6addr
```

NOTE: To enable the ACOS device to send an unsolicited PCP ANNOUNCE message, you must configure the source IPv4 or IPv6 address of the PCP ANNOUNCE packet that is sent by the ACOS device.

- To enable the ACOS device to validate the client nonce in a PCP request.

```
ACOS(config-pcp:11)# check-client-nonce
```

By default, this option is disabled. If this option is enabled, when the PCP request that matches an existing mapping but the nonce value does not match existing value, the ACOS device returns the **NOT_AUTHORIZED** option.

- To enable/disable ACOS processing of MAP FILTER:

```
ACOS(config-pcp:11)# disable-map-filter
```

By default, this option is disabled, which means that ACOS can handle PCP filter option.

When the **filter** option is set in a PCP MAP, the ACOS device checks the inbound session's source address against the filter. If the **disable-map-filter** option is enabled, the ACOS device will not process filter options in the map request.

- After finishing configuration of the template, use the following command at the global configuration level to set the template as the default PCP template:

```
ACOS(config)# cgnv6 pcp default-template11
```

Displaying and Clearing Session Information

- To display PCP Statistics:

```
ACOS# show cgnv6 pcp statistics
PCP Statistics:
---------------
Packets Received                                    0
PCP MAP Request Processing Success (NAT44)          0
PCP MAP Request Processing Success (DS-Lite)        0
PCP MAP Request Processing Success (NAT64)          0
PCP PEER Request Processing Success (NAT44)         0
PCP PEER Request Processing Success (DS-Lite)       0
PCP PEER Request Processing Success (NAT64)         0
PCP ANNOUNCE Request Processing Success (NAT44)     0
PCP ANNOUNCE Request Processing Success (DS-Lite)   0
PCP ANNOUNCE Request Processing Success (NAT64)     0
Packet Not a PCP Request                            0
Packet Too Short                                    0
Response No Route                                   0
Unsupported PCP version                             0
PCP Request Not Authorized                          0
PCP Request Malformed                               0
Unsupported PCP Opcode                              0
Unsupported PCP Option                              0
PCP Option Malformed                                0
No System or NAT Resources                          0
Unsupported Mapping Protocol                        0
User Quota Exceeded                                 0
Cannot Provide Suggested Port When PREFER_FAILURE   0
PCP Client Address Mismatch                         0
Excessive Remote Peers                              0
Packet Dropped For Not Coming From NAT Inside       0
L3/L4 Process Error                                 0
Internal Error                                      0
Unsolicited Announce Sent                           0
Unsolicited Announce Send Failure                   0
HA Sync PCP Epoch Sent                              0
HA Sync PCP Epoch Recv                              0
```

- To clear PCP statistics:

```
ACOS# clear cgnv6 pcp statistics
```

- When displaying full-cone sessions, a "PCP" flag indicates that the full-cone session is created by PCP request:

```
ACOS# show cgnv6 lsn full-cone-sessions
Prot Inside Address   NAT Address   Outbnd Inbnd Pool CPU Age Flags
-------------------------------------------------------------------
TCP  20.1.1.172:2013  10.1.1.9:2013  0      0    1    2  120    -
TCP  20.1.1.172:2014  10.1.1.9:2014  0      0    1    2  600   PCP
TCP  20.1.1.172:2015  10.1.1.9:2015  0      0    1    2  120   -
```

- To display only PCP-created full-cone sessions:

```
ACOS# show cgnv6 lsn full-cone-sessions pcp
Prot Inside Address  NAT Address   Outbnd Inbnd  Pool CPU Age Flags
-------------------------------------------------------------------
TCP  20.1.1.172:2014 10.1.1.9:2014 0      0      1    2   600 PCP
```

- To clear only the PCP-created full-cone sessions:

```
ACOS# clear cgnv6 lsn full-cone-sessions pcp
```

Destination Based NAT

This chapter provides information about how you can match client traffic based on the destination information and override the NAT settings for the matching traffic.

The following topics are covered:

Overview

You can match client traffic based on the destination and complete the following actions:

- Perform Source NAT with a different NAT pool on matching traffic.

  This is only for initial requests.

- Perform Destination NAT and translate the destination address of the matching traffic to an IP address from an IP list or to an IP address which is resolved from a DNS domain.
- Mark the traffic by setting the Diffserv Control Point (DSCP) value in the IP header.
- Pass the traffic through without performing NAT.
- Configure idle timeout values for TCP/UDP/ICMP LSN sessions.
- Drop the traffic.
- Perform one-to-one NAT with a specific NAT pool.

NOTE: TCP/UDP/ICMP idle timeouts configured on an LSN rule-list have higher precedence than the default service port timeout. LSN protocol timeouts can be based on a destination network or on a host address with a service port, so they are more specific matches.

TCP idle timeout is configured independently of TCP half-closed session timeouts. To configure TCP half-closed session timeouts, use the **`cgnv6 lsn half-close-timeout`** command.

You can match based on the following destination information:

- Destination IPv4 host or subnet address
- Destination IPv4 host based on DNS domain-name or domain-list
- Traffic type (ICMP, TCP, UDP, or other)
- Destination TCP or UDP port
- DSCP value

This feature applies to the following types of client traffic:

- CGN/LSN
- NAT64
- NAT44 (Destination NAT based on domain-name)
- DS-Lite
- Fixed-NAT

You can configure the actions and destination matches in rule-lists.

NOTE:

- The option to redirect traffic to a different pool or pool group applies only if the client does not have a NAT session. If the client already has a NAT session, the sticky NAT feature keeps the client on the same NAT address, regardless of the LSN rule-list configuration.
- The one-to-one-snat option is not applicable to NAT64 or DS-Lite. For these features, the option is ignored and the traffic is processed based only on source IP address. (No rule-list is applied.)
- The snat option is not applicable to Fixed-NAT (Fixed-NAT44, Fixed-NAT64 or Fixed-NAT for DS-Lite). For these features, the option is ignored and the traffic is dropped. (For drop statistics, see the "Fixed NAT Dest Rules List Source NAT Drop" counter in the output of the **`show cgnv6 fixed-nat statistics`** command.

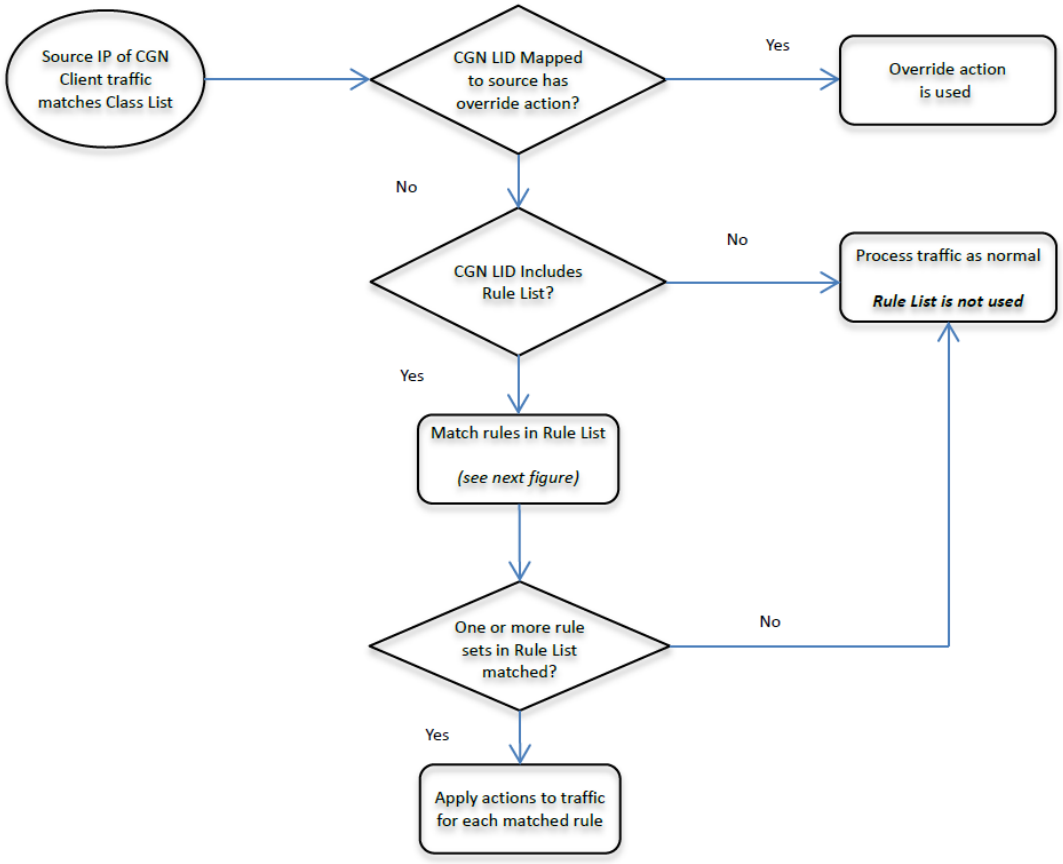
CGN Rule-list Processing Flow

Rule matching for CGN rule-lists occurs in the following way:

1. ACOS determines whether the traffic is eligible for processing using the rule-list. This process depends on the configuration.

   Figure 20 illustrates the initial processing.

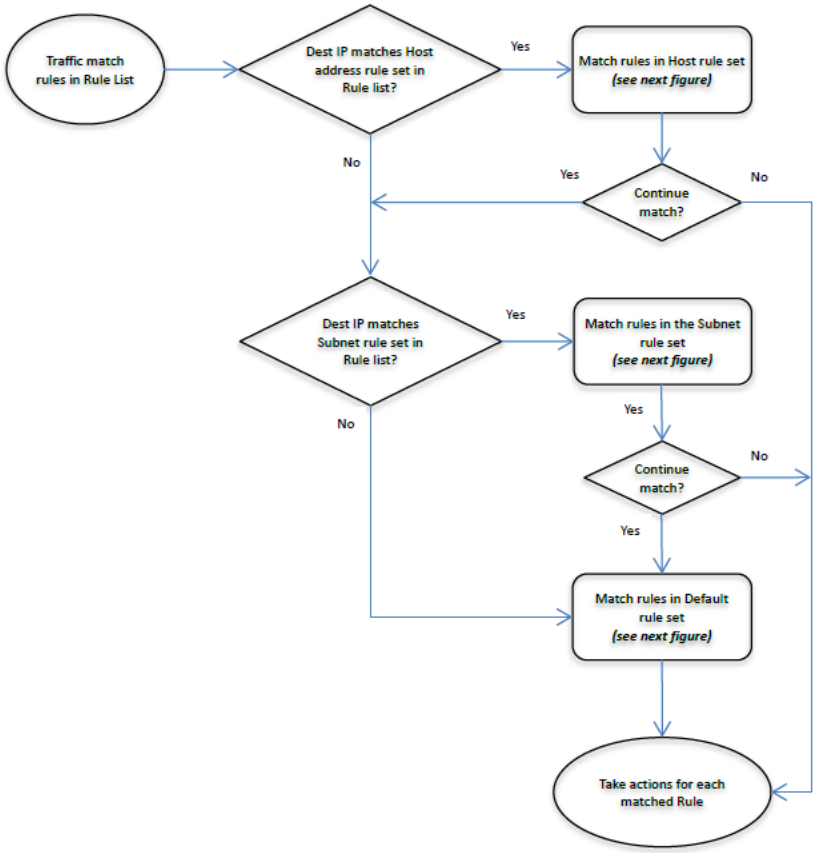
Figure 20 : CGN Rule-list Processing



2. The lists are checked in the following order:
   a. Host list
   b. Subnet list
   c. Default list

   Figure 21 illustrates the matching process for traffic that is eligible for processing by using the rule-list. ACOS checks all applicable entry lists.

Figure 21 : CGN Rule-list Processing



In each list, entries are matched in the following order:

i. By the specific protocol port number.

ii. By the port number of a specific protocol.

iii. By the DSCP value and remark a matching rule has any of the following actions.

3. The following actions can be applied along with other actions:

- dnat domain or ipv4-list
- drop
- one-to-one-snat pool
- pass-through

- set-dscp
- snat pool
- template http-alg

4. The following actions supersede any other actions:

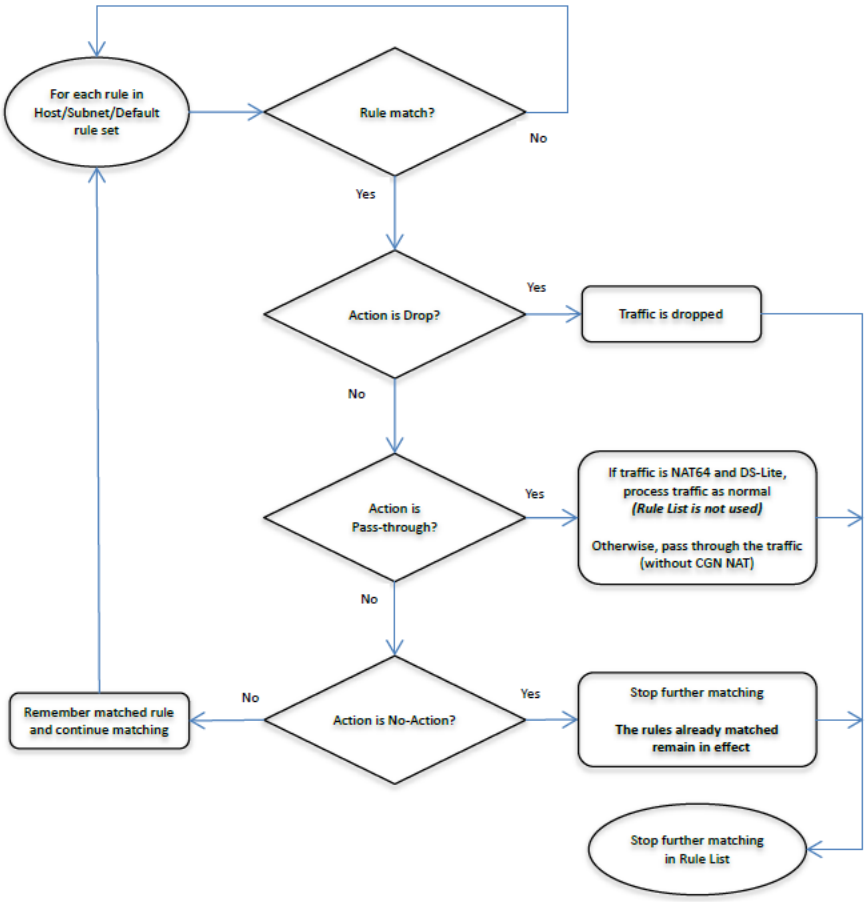
- **Drop?** If traffic matches a drop rule, the traffic is dropped, and rule-list matching stops.
- **Pass-through?** If traffic matches a pass-though rule, the traffic is passed through, and rule-list matching stops.

In either case, no other actions are performed, even if other matching rules have other actions.

- **No-action?** If a rule has this action, matching stops, but all previous matching actions are performed.

If more than one matching rule has the same action, the rule with the most specific match is used. If there are no matches to an action in the rule-list, the traffic is passed. Figure 22 illustrates the processing of the actions. For a complete list of actions, see *Command Line Reference*.

Figure 22 : CGN Rule-list Processing



Configuring Destination Matching

To configure destination matching:

1. Configure an LSN rule-list that specifies the destination information on which to match and the action to perform on matching traffic.

   The destination options are default or an IP address.

2. Add the rule-list to an LID.
3. Add the LSN LID to a class list.

CGN Header Enrichment Matching Domain Names

This feature complements the support of domain name ACL matching for LSN/NAT64/DS-lite/6rd-NAT64/Fixed NAT/One-to-One NAT. The domain name is used to classify traffic if destination IP address matching fails when the ACOS device does not have the same DNS server configurations as the client does.

This feature supports HTTP traffic that contains a domain name in the HTTP request. A new CLI **`http-match-domain-name`** is provided to allow user to enable/disable matching domain name in HTTP requests.

Limitations:

- In the case of multiple domain names for one IP, the actions for these domain names under the same ACL must be configured the same.
- This feature is manly for http-alg traffic, other actions might not work.
- Queue HTTP packets for HTTP host parsing is not supported. This feature only supports HTTP requests with the Host header in the first TCP segment.

Configuring an LSN Rule-list by using the GUI

1. Navigate to **CGN** > **LSN** > **Rule Lists**.
2. Click **Create**.
3. Enter a name for the rule-list.
4. Select the Type:
   - IP Address, Netmask – The rule is used only for the specified IP address or subnet.
   - Default – The rule is used for IP addresses that do not match an IP-specific rule.
   - Domain Name – The rule is used only for the specified domain name.
5. Click **Add Config** and configure rule settings.
6. Click **Create**. Repeat for each rule.
7. Click **Create** to save the rule-list to the running-config.
8. Click **Save** to save the rule-list to the startup-config.

Configuring an LSN Rule-list by using the CLI

To configure an LSN rule-list:

1. Enter this following command to change the CLI to the configuration level for the specified LSN rule set.

```
ACOS(config)# cgnv6 lsn-rule-list55
```

   This command is entered at the global configuration level of the CLI.

2. Enter this command to change the CLI to the configuration level for the specified LSN rule set:

```
ACOS(config-lsn-rule-list)# default
```

   This command enters the configuration level for the default set of rules. The default set of rules is used for traffic that does not exactly match an IP host or subnet rule. (See below.)

3. Enter the following command to enter the configuration level for the set of rules to apply to the specified domain name:

```
ACOS(config-lsn-rule-list)# domain-name www.abc.com
ACOS(config-lsn-rule-list-domain-name)# tcp port 22 action drop
```

4. Enter the following command to enable matching of domain name in the HTTP request:

```
ACOS(config-lsn-rule-list)# http-match-domain-name
```

5. Enter the following command to enter the configuration level for the set of rules to apply to the specified IP host address or subnet:

```
ACOS(config-lsn-rule-list)# ip 1.1.1.1/22
```

6. Enter the following command to configure DSCP marking:

```
ACOS(config-lsn-rule-list-domain-name)# dscp 62 action set-dscp inbound
63
```

   This command completes matches that are based on the DSCP classification in traffic, and marks the DSCP value before forwarding the traffic. For more information, see Quality of Service with DSCP.)

7. Enter the following command to perform the specified action on matching ICMP traffic:

```
ACOS(config-lsn-rule-list-domain-name)# icmp no-action
```

8. Enter the following command to perform the specified action on matching traffic of types other than ICMP, TCP, or UDP:

```
ACOS(config-lsn-rule-list-domain-name)# others no-action
```

9. Enter the following commands to perform the specified action on matching traffic with the specified TCP or UDP port(s):

```
ACOS(config-lsn-rule-list-domain-name)# tcp port 1 no-action
ACOS(config-lsn-rule-list-domain-name)# udp port 1 no-action
```

(For information about the actions you can specify, see the *Command Line Reference* guide.)

NOTE: The **no-action** option excludes the matching traffic from the actions in the rule-list but still performs NAT for the traffic.

Adding the LSN Rule-list to an LSN LID

Enter the following command to add the LSN rule-list to the LSN LID:

```
ACOS(config)# cgnv6 lsn-lid 22
ACOS(config-lsn-lid)# lsn-rule-list destination 55
```

Adding the LSN LID to a Class List

Enter the following command to specify the source IP address on which to match and add the LSN LID to a class list:

```
ACOS(config)# class-list 11
ACOS(config-class-list)# 1.1.1.1/22 lsn-lid 22
```

Destination NAT

Destination NAT replaces the destination IP address of matching client-to-server traffic with an IP address either from an IP list or an IP address which is resolved from a DNS domain.

This option is useful in cases where you want to allow the client traffic, but send it to a different destination IP address. For example, if some client traffic initially is addressed to an incorrect gateway, you can correct the gateway address using this feature.

ADP Support

Destination NAT using rule-lists is supported in private partitions, with the rule-list and IP list in the same partition.

Configuring Destination NAT with LSN-Rule-List

There are two approaches to configure destination NAT with LSN-Rule-List:

- Configure destination NAT using IP list – To use IP address from an IP list as destination NAT address.
- Configure destination NAT using domain-name – To use IP address associated with a domain-name as destination NAT address.

Configuring destination NAT using IP list

Configuring an IP List

To configure an IP list by using the CLI:

1. Enter the following command to create the IP list and access the configuration level for this list:

```
ACOS(config)# ip-list 55
```

2. Enter the following command to specify the NAT IP addresses or address range:

```
ACOS(config-ip-list)# 1.1.1.1 to 2.2.2.2
```

Configuring a LSN-Rule-list

A lsn-rule-list includes rules to apply DNAT action using IP list.

NOTE: ACOS does not perform destination NAT if the original destination IP address is not reachable from the ACOS device.

To configure a lsn-rule-list:

1. Enter the following command:

```
ACOS(config)# cgnv6 lsn-rule-list 66
```

This command is entered at the global configuration level to create the rule-list and access the configuration level for it.

2. Enter the following command to access the configuration level for rule sets in the list:

```
ACOS(config-lsn-rule-list)# ip 1.1.1.0/24
```

Configuring DNAT rules

To configure DNAT rules:

1. At the configuration level for each rule set, enter the following commands to configure the destination NAT for CGN:

```
ACOS(config-lsn-rule-list-ip)# tcp port 22 action dnat ipv4-list 55
ACOS(config-lsn-rule-list-ip)# udp port 33 action dnat ipv4-list 20
ACOS(config-lsn-rule-list-ip)# icmp action dnat ipv4-list 30
ACOS(config-lsn-rule-list-ip)# others action dnat ipv4-list 40
```

or

```
ACOS(config-lsn-rule-list-ip)# default action dnat ipv4-list 40
```

The **tcp** and **udp** commands match the traffic based on the destination TCP or UDP port. The **icmp** command matches on the ICMP traffic. The **other** command matches on all other traffic.

The **default** command matches all traffic.

Applying LSN-Rule-List

1. Enter the following command to apply the lsn-rule-list to lsn-lid:

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn-lid)# lsn-rule-list destination 66
```

Configuring destination NAT using domain-name

Configuring a DNS server

The DNS server is used to resolve a domain-name to an IP address.

To configure a DNS server by using the CLI:

1. Enter the following commands to configure the DNS Server:

```
ACOS(config)# ip dns primary 192.168.51.6
```

Configuring a LSN-Rule-list

A lsn-rule-list includes rules to apply DNAT action using domain-name.

To configure a lsn-rule-list:

1. Enter the following command:

```
ACOS(config)# cgnv6 lsn-rule-list 66
```

   This command is entered at the global configuration level to create the rule-list and access the configuration level for it.

2. Enter the following command to access the configuration level for the list rule sets in order to bind the fake IP address with the rule list:

```
ACOS(config-lsn-rule-list)# ip 1.1.1.0/24
```

Configuring DNAT rules

To configure DNAT rules:

1. At the configuration level for each rule set, enter the following commands to configure the destination NAT for CGN:

```
ACOS(config-lsn-rule-list-ip)# tcp port 0 action dnat domain a10rocks.com
ACOS(config-lsn-rule-list-ip)# udp port 0 action dnat domain a10rocks.com
ACOS(config-lsn-rule-list-ip)# icmp action dnat domain a10rocks.com
ACOS(config-lsn-rule-list-ip)# others action dnat domain a10rocks.com
```

or

```
ACOS(config-lsn-rule-list-ip)# default action dnat domain a10rocks.com
```

The **default** command matches all traffic.

Applying LSN-Rule-List

1. Enter the following command to apply the lsn-rule-list to lsn-lid:

```
ACOS(config)# cgnv6 lsn-lid 1
```

```
ACOS(config-lsn-lid)# lsn-rule-list destination 66
```

Verifying if domain-name is resolved successfully

1. Enter the following command to verify if the domain-name is resolved successfully:

```
ACOS(config)# show ip dns-cache
The items of domain and IP in cache
FQDN            IP                  TTL      query-interval
--------------------------------------------------------------
a10rocks.com    7.7.7.212           900      600
```

This command output shows the real IP address of the domain name from the DNS server cache.

Quality of Service with DSCP

ACOS supports the following functionality:

- Layer 3 classification, which reads the value in the Diffserv Control Point (DSCP) field in the IP headers of matching CGN traffic.
- Marking, which sets this value before traffic is forwarded.

  This option provides QoS per-hop behavior (PHB) for CGN traffic.

ACOS automatically performs classification on the first client-to-server packet of a session. You can configure DSCP marking on classified traffic. For each classified session, you can specify the traffic direction to which the marking applies:

- Outbound – Client-to-server traffic
- Inbound – Server-to-client traffic

Supported DSCP Markings

You can set the DSCP value for inbound or outbound traffic to one of the values in Table 10.

Table 10 : Supported DSCP values

Type of DSCP Value	Supported Values[1]
DSCP number	0-63
Assured Forwarding (AF)	AF11, AF12, AF13 AF21, AF22, AF23 AF31, AF32, AF33 AF41, AF42, AF43
Class Selector (CS)	CS1-CS7

Consider the following information:

- Rule matching is performed on the first packet for each session. The DSCP action is taken for each packet in the same session.
- ACOS performs marking but does not perform QoS actions based on DSCP values.
- For tunneled traffic (IPv4-in-IPv6 or IPv6-in-IPv4), ACOS performs marking on both packets, the outer packet and the encapsulated packet.

Configuring DSCP Marking for CGN

To configure DSCP marking for CGN:

1. Configure a rule-list that specifies the destination (server) addresses and protocols or ports.
2. Configure an LID that binds to the rule-list.
3. Configure a class list that specifies the source (client) IP addresses, and associates them with the LID bound to the rule-list.

Rule Matching

Matching occurs in the following order, from most granular match to least granular match:

[1]For a list of the binary values for each of these options, see the online help. The CLI does not accept the binary values, but they are listed for reference.

NOTE: The matching rules in CGN Rule-list Processing Flow apply.

- Host rule set – Rule set for a specific IPv4 destination address
- Subnet rule set – Rule set for all hosts in a specific IPv4 destination subnet that do not match a host rule set
- Default rule set – Rule set for all destination hosts that do not match a host or subnet rule set

In each type of rule set, you can configure individual rules to match on and remark the following types of traffic:

- TCP traffic (specific port number, range, or any)
- UDP traffic (specific port number, range, or any)
- ICMP traffic (any port number)
- "Other" traffic (any port number for any traffic other than TCP, UDP, or ICMP)

You also can configure rules to match on the initial DSCP value (the value observed during classification), and to remark matching traffic.

ADP Support

You can configure CGN Rule-list and DSCP marking separately in different partitions.

Configuring the Rule-list

Enter the following command to create the rule-list and access the configuration level for the list:

```
ACOS(config)# cgnv6 lsn-rule-list rule1
```

Accessing the Configuration Level for a Rule Set

Enter the following commands to access the configuration level for rule sets in the list:

```
ACOS(config-lsn-rule-list)# default
```

```
ACOS(config-lsn-rule-list)#  domain-name  2
ACOS(config-lsn-rule-list)# ip  1.1.1.1/11
ACOS(config-lsn-rule-list-ip)#
```

Configuring Rules

- At the configuration level for each rule set, enter the following command to configure a DSCP marking rule for a TCP port or range of ports:

```
ACOS(config-lsn-rule-list-ip)# tcp port 1 action set-dscp  inbound63
ACOS(config-lsn-rule-list-ip)# tcp port 1 action set-dscp  outbound 60
```

- Enter the following command to configure a DSCP marking rule for a UDP port or range of ports:

```
ACOS(config-lsn-rule-list-ip)# udp port 1 action set-dscp  inbound63
ACOS(config-lsn-rule-list-ip)# udp port 1 action set-dscp  outbound 60
```

- Enter the following command to configure a DSCP marking rule for ICMP traffic:

```
ACOS(config-lsn-rule-list-ip)# icmp action set-dscp  inbound63
ACOS(config-lsn-rule-list-ip)# icmp action set-dscp  outbound 60
```

- Enter the following command to configure a DSCP marking rule for other types of traffic:

```
ACOS(config-lsn-rule-list-ip)# otheraction set-dscp  inbound63
ACOS(config-lsn-rule-list-ip)# other  action set-dscp  outbound 60
```

- Enter the following command to match based on DSCP classification:

```
ACOS(config-lsn-rule-list-ip)# dscp any action set-dscp  inbound63
ACOS(config-lsn-rule-list-ip)# other  action set-dscp  outbound 60
```

Configuring the LID

1. Enter the following command to create the rule-list and access the configuration level for the list:

```
ACOS(config)# cgnv6 lsn-lid  11
```

2. Enter the following command to bind the rule-list to the LID:

```
ACOS(config-lsn-lid)# lsn-rule-list rule1
```

Configuring the Class List

1. Enter the following command to create the class list and access the configuration level for the list:

```
ACOS(config)# class-list  33
```

2. Enter the following command to specify a client host or subnet address and associate the LID with the host or address:

```
ACOS(config-class-list)# 1.1.1.1/22 lsn-lid 11
```

Destination Rule Support for Fixed-NAT

You can use a CGN rule-list in a Fixed-NAT configuration.

Overview

You can use actions in the rule-list to process client traffic before assigning a Fixed-NAT mapping of the outside address to the client. For example, you can use a CGN rule-list to perform Destination NAT or DSCP marking for Fixed-NAT client traffic before forwarding the traffic.

Configuring Rule Support for Fixed-NAT

To configure CGN rule-list processing for Fixed-NAT clients:

1. Configure the CGN rule-list entries. See Configuring the Rule-list.
2. Configure IP lists for the inside and NAT addresses (unless you plan to specify them when you perform the following step). See Configuring the IP Lists.
3. Enable Fixed-NAT. See Enabling Fixed-NAT.

Configuring the IP Lists

The following commands create the IP list and configure the client or NAT IP addresses or address range:

```
ACOS(config)# ip-list 33
ACOS(config-ip-list)# 1.1.1.1 to 1.1.1.10
ACOS(config-ip-list)# exit
ACOS(config)# ip-list 44
ACOS(config-ip-list)# 2.1.1.1 to 2.1.1.10
ACOS(config-ip-list)# exit
```

Enabling Fixed-NAT

Enter the following command to enable Fixed-NAT:

```
ACOS(config)# cgnv6 fixed-nat inside ip-list 33 nat ip-list 44 dest-rule-list rule1
```

Configuration Examples

The following information contains some examples of rule-list configurations.

Single Action

The following rule-list performs source NAT using pool “pool2”, for traffic from client 1.1.1.1 to any destination TCP port at subnet 123.1.1.x.

The following commands configure the rule-list:

```
ACOS(config)# cgnv6 lsn-rule-list rl1
ACOS(config-lsn-rule-list)# ip 123.1.1.0/24
ACOS(config-lsn-rule-list-ip)# tcp port 0 action snat pool pool2
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# exit
```

The following commands configure the LID:

```
ACOS(config)# cgnv6 lsn-lid 1
```

```
ACOS(config-lsn lid)# lsn-rule-list destination rl1
ACOS(config-lsn lid)# exit
```

The following commands configure the class list:

```
ACOS(config)# class-list 1
ACOS(config-class list)# 1.1.1.1/32 lsn-lid 1
```

Multiple Actions

The following rule-list has multiple actions that can be applied to matching traffic. For example, all the following actions are performed on traffic to destination 123.1.1.1:80:

- Process the traffic based on HTTP-ALG template “alg2”.
- Perform source NAT using pool “pool2”.
- Mark outbound traffic with DSCP value 3.

NOTE: Only some actions can be applied together to the same traffic. (See CGN Rule-list Processing Flow.)

The following commands configure the rule-list:

```
ACOS(config)# cgnv6 lsn-rule-list rl2
ACOS(config-lsn-rule-list)# ip 123.1.1.0/24
ACOS(config-lsn-rule-list-ip)# tcp port 80 action template http-alg alg2
ACOS(config-lsn-rule-list-ip)# tcp port 0 action snat pool pool2
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# default
ACOS(config-lsn-rule-list-default)# tcp port 0 action set-dscp outbound 3
ACOS(config-lsn-rule-list-default)# exit
ACOS(config-lsn-rule-list)# exit
```

The following commands configure the LID:

```
ACOS(config)# cgnv6 lsn-lid 2
ACOS(config-lsn lid)# lsn-rule-list destination rl2
ACOS(config-lsn lid)# exit
```

The following commands configure the class list:

```
ACOS(config)# class-list 2
```

```
ACOS(config-class list)# 1.1.1.1/32 lsn-lid 2
```

Drop

The rule-list in this example applies to traffic from client 1.1.1.1. DSCP marking is performed for outbound traffic to destination 123.1.1.x, to any TCP port except 1234.

All traffic to TCP port 1234, at any destination IP address, is dropped.

Since no other actions can be applied along with the action, the traffic is dropped but DSCP marking is not performed.

The following commands configure the rule-list:

```
ACOS(config)# cgnv6 lsn-rule-list drop1234
ACOS(config-lsn-rule-list)# ip 123.1.1.0/24
ACOS(config-lsn-rule-list-ip)# tcp port 0 action set-dscp outbound 3
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# default
ACOS(config-lsn-rule-list-default)# tcp port 1234 action drop
ACOS(config-lsn-rule-list-default)# exit
ACOS(config-lsn-rule-list)# exit
```

The following commands configure the LID:

```
ACOS(config)# cgnv6 lsn-lid 2
ACOS(config-lsn lid)# lsn-rule-list destination drop1234
ACOS(config-lsn lid)# exit
```

The following commands configure the class list:

```
ACOS(config)# class-list 3
ACOS(config-class list)# 1.1.1.1/32 lsn-lid 2
```

No-action

The rule-list in this example applies to traffic from client 1.1.1.1. If traffic matches a rule that has the **no-action** action, processing stops, and none of the rule-list actions are applied. This is true even if the traffic matches a drop rule after matching the no-action rule.

In this example, traffic to destination 123.1.1.8:1234 matches all the rules in the list, including the drop rule. However, no-action stops further matching of the drop rule. Thus, the traffic is marked as DSCP and not being dropped.

The following commands configure the rule-list:

```
ACOS(config)# cgnv6 lsn-rule-list drop1234
ACOS(config-lsn-rule-list)# ip 123.1.1.8/32
ACOS(config-lsn-rule-list-ip)# tcp port 0 action set-dscp outbound 2
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# ip 123.1.1.0/24
ACOS(config-lsn-rule-list-ip)# tcp port 0 no-action
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# default
ACOS(config-lsn-rule-list-default)# tcp port 1234 action drop
ACOS(config-lsn-rule-list-default)# exit
ACOS(config-lsn-rule-list)# exit
```

The following commands configure the LID:

```
ACOS(config)# cgnv6 lsn-lid 2
ACOS(config-lsn lid)# lsn-rule-list destination drop1234
ACOS(config-lsn lid)# exit
```

The following commands configure the class list:

```
ACOS(config)# class-list 3
ACOS(config-class list)# 1.1.1.1/32 lsn-lid 2
```

Destination NAT

The rule-list in this example configures destination NAT for traffic sent to any TCP port at destination IP address 1.2.3.4. For TCP requests sent to that address, ACOS changes the destination IP address to an address in the IP list, 158.1.1.2 or 158.1.1.3.

To begin, the following commands configure an IP list containing the NAT addresses to use:

```
ACOS(config)# ip-list DNAT_LIST
ACOS(config-ip list)# 158.1.1.2
ACOS(config-ip list)# 158.1.1.3
```

The following commands configure the rule-list:

```
ACOS(config)# cgnv6 lsn-rule-list RLIST
ACOS(config-lsn-rule-list)# ip 1.2.3.4/32
ACOS(config-lsn-rule-list-ip)# tcp port 0 action dnat ipv4-list DNAT_LIST
```

DSCP Marking (Example 1)

The commands in this example deploy a simple DSCP marking configuration, for sessions between client 1.1.1.1 and server 5.5.5.5 (any TCP port). Traffic is marked as follows:

- Outbound (client-to-server) – DSCP value 20
- Inbound (server-to-client) – DSCP value 10

The following commands configure the rule-list:

```
ACOS(config)# cgnv6 lsn-rule-list 1
ACOS(config-lsn-rule-list)# ip 5.5.5.5/32
ACOS(config-lsn-rule-list-ip)# tcp port 0 action set-dscp inbound 10
ACOS(config-lsn-rule-list-ip)# tcp port 0 action set-dscp outbound 20
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# exit
```

The following commands configure the LID:

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn lid)# lsn-rule-list destination 1
ACOS(config-lsn lid)# exit
```

The following commands configure the class list:

```
ACOS(config)# class-list 1
ACOS(config-class list)# 1.1.1.1/32 lsn-lid 1
```

DSCP Marking (Example 2)

The following commands configure a rule-list to perform the following DSCP marking for traffic from client 1.1.1.2:

- Destination 123.1.1.8:80 – Mark with DSCP value 3, for both outbound (client-to-server) and inbound (server-to-client) traffic.
- Destination 123.1.1.x, any TCP port – Mark with DSCP value af11, for both outbound and inbound traffic.
- Any destination, with DSCP value cs1 – Mark with DSCP value 0, for both outbound and inbound traffic.

The following commands configure the rule-list:

```
ACOS(config)# cgnv6 lsn-rule-list qos1
ACOS(config-lsn-rule-list)# ip 123.1.1.8/32
ACOS(config-lsn-rule-list-ip)# tcp port 80 action set-dscp inbound 3
ACOS(config-lsn-rule-list-ip)# tcp port 0 action set-dscp outbound 3
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# ip 123.1.1.0/24
ACOS(config-lsn-rule-list-ip)# tcp port 0 action set-dscp inbound af11
ACOS(config-lsn-rule-list-ip)# tcp port 0 action set-dscp outbound af11
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# default
ACOS(config-lsn-rule-list-default)# dscp cs1 action set-dscp inbound 0
ACOS(config-lsn-rule-list-default)# dscp cs1 action set-dscp outbound 0
ACOS(config-lsn-rule-list-default)# exit
ACOS(config-lsn-rule-list)# exit
```

The following commands configure the LID:

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn lid)# lsn-rule-list destination qos1
ACOS(config-lsn lid)# exit
```

The following commands configure the class list:

```
ACOS(config)# class-list 1
ACOS(config-class list)# 1.1.1.2/32 lsn-lid 1
```

DSCP Marking (Example 3)

The rule-list in this example applies different settings for the same option (in this case, DSCP marking). When more than one rule applies the same action, but with

different values, the rule with the most specific match is used. This rule-list applies to traffic from client 1.1.1.3.

In this example, traffic is marked as follows:

- Destination 123.1.1.8:80 – Mark with DSCP value 1, for outbound (client-to-server) traffic.
- Destination 123.1.1.8, any TCP port – Mark with DSCP value 2, for outbound traffic.
- Destination 123.1.1.x, any TCP port – Mark with DSCP value 3, for outbound traffic.
- Any other destination IP address and any TCP port – Mark with DSCP value 4, for outbound traffic.

The following commands configure the rule-list:

```
ACOS(config)# cgnv6 lsn-rule-list qos-tcp
ACOS(config-lsn-rule-list)# ip 123.1.1.8/32
ACOS(config-lsn-rule-list-ip)# tcp port 80 action set-dscp outbound 1
ACOS(config-lsn-rule-list-ip)# tcp port 0 action set-dscp outbound 2
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# ip 123.1.1.0/24
ACOS(config-lsn-rule-list-ip)# tcp port 0 action set-dscp outbound 3
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# default
ACOS(config-lsn-rule-list-default)# tcp port 0 action set-dscp outbound 4
ACOS(config-lsn-rule-list-default)# exit
ACOS(config-lsn-rule-list)# exit
```

The following commands configure the LID:

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn lid)# lsn-rule-list destination qos-tcp
ACOS(config-lsn lid)# exit
```

The following commands configure the class list:

```
ACOS(config)# class-list 1
ACOS(config-class list)# 1.1.1.3/32 lsn-lid 1
```

Based on these rules, here are the DSCP markings performed for some matching traffic:

- Traffic to 123.1.1.8:80 – Outbound marked with DSCP 1.
- Traffic to 123.1.1.8:123 – Outbound marked with DSCP 2.
- Traffic to 123.1.1.1:80 – Outbound marked with DSCP 3.
- Traffic to 123.1.1.1:123 – Outbound marked with DSCP 4.

Disabling Source NAT for Destination NAT Action in the Rule-List

Client traffic can be processed basing solely on the destination NAT matching rules. Only destination NAT rule-matching can be configured. If source NAT matching is disabled for a given destination NAT, then if client traffic matches both a source rule and the specific destination NAT, then only the destination NAT rule action will be used.

This LSN rule-list action for destination NAT only supports outbound traffic. The intended use is for when the configured destination NAT sessions only are for outbound traffic on the ACOS device. Once the traffic is routed to the server, after the destination NAT action is taken, then the server should send the return traffic to bypass the ACOS device.

Disabling source NAT rule matching is only supported for NAT44 and not for ALGs. It is not possible to disable source NAT rule matching for Fixed-NAT, as Fixed-NAT requires the source NAT. Additionally, disabling the source NAT means that CGN logging will not be triggered for the given traffic as there is no port mapping.

Configuring Destination NAT Action Only

To configure rule matching action on a destination NAT only, enter the following commands at the LSN rule-list configuration level, depending on the type of traffic to disable source NAT.

Below is a configuration example disabling source NAT action for ICMP traffic for the rule-set of destination IP addresses 10.1.2.0 /24. Source NAT action is also disable for TCP traffic for the rule-set of destination IP addresses 10.1.3.0 /24.

```
ACOS(config)# ip-list dst-list
ACOS(config-ip-list)# 10.2.3.4 to 102.2.3.5
ACOS(config-ip-list)# exit
ACOS(config)# cgnv6 lsn-rule-list
ACOS(config-lsn-rule-list)# ip 10.1.2.0/24
```

```
ACOS(config-lsn-rule-list-ip)# icmp action dnat ipv4-list dst-list
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# ip 10.1.3.0/24
ACOS(config-lsn-rule-list-ip)# tcp port 80 action dnat ipv4-list dst-list
```

Displaying and Clearing Rule-list Information

- Enter the following command to display rule-list information:

```
ACOS(config)# show cgnv6 lsn-rule-list33statistics
```

- Enter the following command to clear rule-list statistics:

```
ACOS(config)# clear cgnv6 lsn-rule-list all
```

Attack Detection and Mitigation

This chapter provides an overview of DDoS mitigation for IPv6 Migration and illustrates how to configure IP blacklisting, IP anomaly filtering, connection rate limiting, and so on for limiting protocol attacks and volumetric attacks that consume server resources.

The following topics are covered:

Overview

ACOS provides security protection to help mitigate against some forms of Distributed Denial of Service (DDoS) attacks on servers. Some of the features such as IP blacklisting, IP anomaly filtering, connection rate limiting, and more aim to limit protocol attacks and volumetric attacks that consume server resources.

Protocol attacks consist of packets that invalidly formed or contain protocol abnormalities. These attacks are meant to exploit a protocol feature or bug in order to consume server resources. Typically, a resource attack is conducted by directing a high rate of invalid traffic toward the target system, to overwhelm the system's resources.

Volumetric attacks are brute-force assaults, often launched using botnets, that attempt to consume as many network resources as possible on the target system. This type of attack can be used not only to disrupt service, but also to provide a diversion for more nefarious and targeted network intrusion, such as identity theft.

The following are the DDoS Identification and Mitigation features for CGN:

- IP Blacklist for DDoS Protection
- IP Blacklist for DDoS Protection
- Selective Filtering for LSN
- IP Anomaly Filtering
- Connection Rate Limiting
- Reduced CPU Overhead for CPU Round Robin
- SYN Cookie

IP Blacklist for DDoS Protection

ACOS offers more nuanced DDoS protection by being able to blacklist individual IP addresses on a DDoS attack in a NAT IP pool.

When a DDoS attack targeted towards a specific IP address within a NAT IP pool is detected, then the ACOS device will add that IP address to the blacklist. The ACOS blacklist can contain up to 1024 IP addresses at any given moment.

ACOS determines a DDoS attack when a large number of out-of-state packets (initiated from the internet) are sent to a NAT IP within a short time. The packets-per-second (PPS) limit configures the maximum number of out-of-state packets allowed before an IP is blacklisted. The PPS threshold can range from 0 to 30000000 out-of-state packets. Out-of-state packets include the first packet for new sessions or illegitimate packets that do not match the session. Packets that do not match LSN full-cone sessions are also considered out-of-state. However, if **`include-existing-session`** is enabled, the packets that match the forward tuple of a session initiated from the internet (i.e., an inbound session) are also included.

ACOS detects a DDoS attack when the PPS per NAT IP or per NAT IP port exceeds the configured threshold limit. After detecting the attack, one of the following actions can be taken:

- **Log**—Log the event
- **Drop**—Drop the packets, log the event, and mark the NAT IP as unusable
- **Redistribute Route Map**—Have an upstream router drop or redirect the packets and log the event. Configuring this option enables the Remotely Triggered Black Hole (RTBH) technique for DDoS protection. For more information, see Remotely Triggered Black Hole.

By default, DDoS NAT IP logging is enabled and the event is logged for all actions. The logged event can be viewed using a **`show`** command.

When a NAT IP address is marked as unusable, ACOS clears off all the sessions using that NAT IP address. While the sessions are being cleared, the internal client mapped to the NAT IP experiences a short service interruption. Later, the internal client is assigned to a different NAT IP address, allowing traffic to resume normal flow. The NAT IP address remains unusable for all internal clients until it is removed from the blacklist.

NOTE: If the rate of out-of-state packets drops below the threshold, but exceeds the threshold again before the configured expiration, then the expiration time will be reset back to the maximum time.

When a NAT IP is removed from the blacklist and ACOS restores the NAT IP to the NAT pool, then the NAT IP is free to be used by any internal client. The original internal client mapped to that NAT IP is not necessarily restored back to that given NAT IP.

Remotely Triggered Black Hole

Remotely Triggered Black Hole (RTBH) is a routing technique that provides DDoS protection by dropping malicious traffic before it enters a protected network.

When a DDoS attack targeted towards a specific NAT IP address is detected, the victim NAT IP prefix is marked with configured Border Gateway Protocol (BGP) community that advertises the NAT IP prefix to the edge routers. The edge routers route all traffic coming towards the NAT IP to a null route or a black hole i.e., the edge routers drop all traffic unconditionally from the network, thereby mitigating the DDoS attack.

The RTBH protection technique can be configured using the **redistribute-route** action.

```
ACOS(config)# cgnv6 ddos-protection packets-per-second ip 100000 action redistribute-route mymap
```

Here, *mymap* is the route map that must be specified for routing or redirecting the traffic

When the RTBH comes into effect:

- The victim NAT IP is marked unusable.
- All existing sessions that use this NAT IP address are cleared and new sessions are not allowed to use this NAT IP.
- The NAT IP is added to a blocklist, as a result, all traffic directed to the NAT IP is unconditionally dropped and the NAT IP is inaccessible.

However, in certain scenarios, it may be necessary to access the victim NAT IP. Consider a case where many subscribers use a NAT IP to communicate with the outside servers. In case of a DDoS attack, when the RTBH is triggered, all traffic directed to the NAT IP is dropped unconditionally. Although the attack is mitigated, all subscriber connections are also interrupted since the NAT IP is inaccessible. In

such situations, you can specify the **forward** option while configuring the RTBH action.

```
ACOS(config)# cgnv6 ddos-protection packets-per-second ip 100000 action
redistribute-route mymap forward
```

When this option is configured, BGP advertises the NAT IP to the upstream router. However, in this case, ACOS forwards the traffic (instead of dropping it unconditionally), thereby allowing subscribers to access the NAT IP under attack even when RTBH is in effect. Additionally, the NAT IP can be used for new sessions.

Additional parameters such as **expiration**, **remove-wait-timer**, and **timer-multiply-max** can also be configured.

These parameters govern the overall flow of the following events:

1. A timer, also known as a blackhole timer (configured using the **expiration** parameter), determines the duration the blackhole route must be disabled when a DDoS attack is detected. When this timer expires, ACOS removes the blackhole route remotely without enabling the blackholed NAT IP and applies a wait period of five minutes (configured using the **remove-wait** parameter).
2. During the remove-wait period, if ACOS detects the attack again, it re-initiates the blackhole entry and extends the blackhole timer by a duration that is twice the expiration time (for the first time).
3. The second time, the duration is thrice the expiration time and this duration continues to increment until it reaches the maximum limit specified by the **timer-multiply-max** parameter.
4. If ACOS does not detect attacks during the remove-wait period, it removes the blackhole entry completely and starts forwarding packets to the NAT IP address.

Configuring NAT IP blacklisting for DDoS Protection

This section provides instructions on configuring the packets-per-second limit, action to be performed when the number of packet-per-second limit is crossed, expiration time, and time-multiply for extending the black hole timer.

Perform the following:

1. Configure the maximum number of out-of-sequence packets allowed to be sent to one NAT IP per second. In this example, the maximum number of out-of-sequence packets allowed is 10000

```
ACOS(config)# cgnv6 ddos-protection packets-per-second ip 10000
```

2. Configure the action to be performed. Three types of actions can be performed:

   - **Log**—In this example, when the packets-per-second rate is over 10000, an event is logged. The NAT IP continues to send traffic as normal.

```
ACOS(config)# cgnv6 ddos-protection packets-per-second ip 10000
action log
```

   - **Drop**—In this example, when the packets-per-second rate is over 25000, an event is logged and ACOS drops the incoming packets. The NAT IP is marked as unusable and the existing sessions are cleared.

```
ACOS(config)# cgnv6 ddos-protection packets-per-second ip 25000
action drop
```

   - **Redistribute Route Map**—In this example, when the packets-per-second rate is over 100000, the information is redistributed via BGP to an upstream router with the route-map named “map-1.” The route-map is inserted into the existing route-map sequence at 100. The community number is set as a hint from the ACOS device for the upstream router.

     In this example, map-1 is configured as the route-map and then inserted into the existing route-map sequence at 100.

```
ACOS(config)# route-map map-1 permit 100
ACOS(config-route-map:100)# set community 33:44
ACOS(config-route-map:100)# exit
ACOS(config)# cgnv6 ddos-protection packets-per-second ip 100000
action redistribute-route map-1
```

     After configuring map-1 as the route map, configure the upstream router as shown in the following example:

```
upstream-router(config)# ip community-list standard COM-1 permit
33:44
upstream-router(config)# route-map map-1 permit 100
upstream-router(config-route-map)# match community COM-1
```

```
upstream-router(config-route-map)# set ip next-hop 9.9.9.9
upstream-router(config-route-map)# exit
upstream-router(config)# router bgp 123
upstream-router(config-router)# neighbor 1.1.1.1 route-map map-1 in
```

Notice that the route-map community number matches the number configured on the ACOS device. When the community number matches, then the configured next-hop action is taken. The last two lines of the router configuration create the filter for route advertisements received from the ACOS device.

3. Configure the expiration time for ACOS to revert the action after pps is decreased below the threshold level. The default expiration value is 3600 seconds. In this example, the expiration is set to 100 seconds.

```
ACOS(config)# cgnv6 ddos-protection packets-per-second ip 100000 action
redistribute-route map-1 expiration 100
```

4. For Redistribute Route Map only, configure the maximum value of the timer multiplier for DDoS attacks that last longer. The timer multiplier can be configured to a value between 1 to 100. Default value is 6. In this example, the maximum value of the time multiplier is 7.

```
ACOS(config)# cgnv6 ddos-protection packets-per-second ip 100000 action
redistribute-route map-1 expiration 100 timer-multiply-max 7
```

Use the **show cgnv6 ddos-protection ip-entries** command to view the blacklisted NAT IP addresses as well as their current packet rate and expiration time.

```
ACOS(config)# show cgnv6 ddos-protection ip-entries
Address               PPS          Expiration  Hints
-------------------------------------------------------------
11.1.1.1              653000       700         Timer-Multiplier 7
```

When the out-of-sequence packets are lower than pps limit and the time expires, ACOS moves to the Remove-Wait state as shown in this example.

```
ACOS(config)# show cgnv6 ddos-protection ip-entries
Address               PPS           Expiration    Hints
-------------------------------------------------------------
11.1.1.1              653000        300           Remove-Wait
```

The following example displays output for the command show ddos-protection ip-entries all, which displays the status of all NAT IP addresses, their configured packets-per-second threshold, and their expiration time.

```
ACOS# show cgnv6 ddos-protection ip-entries all
(*) L4 PPS Threshold Exceeded
(**) L3 PPS Threshold Exceeded

Address              PPS         Expiration   L4-Entries
-------------------------------------------------------
11.1.1.2(*)          19400       -            1
11.1.1.1(**)         844000      60           3
11.1.1.7              0          -            0
```

The last column, “L4-Entries”, is the count of Layer 4 ports detected to be under DDoS attack for the given NAT IP. This is the number of ports which have exceeded the configured Layer 4 packets-per-second threshold for the given NAT IP.

The single asterisk (*) indicates that one or more Layer 4 port for the respective NAT IP has exceeded the Layer 4 packets-per-second. Double asterisks (**) indicate that the Layer 3 packets-per-second threshold for the respective NAT IP has been exceeded.

NAT IP Black-holing via BGP

IP addresses are black-listed upon receiving BGP route advertisements.

ACOS device will disable a specific NAT IP from the NAT pool. When a NAT IP is disabled, all existing sessions using this NAT IP will be cleared; new sessions will stop using this NAT IP. Conversely, upon receiving a BGP route withdrawal, ACOS device will resume the specific NAT IP previously disabled.

NOTE: The BGP neighbor may only be configured in the shared partition. However, the black holed NAT IPs may belong to the shared or L3V partitions.

When an IP is black-listed, upon receiving a BGP route update, ACOS device can disable a specific NAT IP fallen into the configured BGP DDoS zone.

NOTE: Only BGP route updates having the netmask of /32 can be added to Blackhole lists.

To disable a NAT IP based on BGP advertisement from an upstream router, enter the **cgnv6 ddos-protection disable-nat-by bgp zone** command at the configuration level.

After configuring a neighboring BGP router, route updates from this neighboring router is treated specially using the **acos-application-only** command. This configuration must be configured on the ACOS device to disable or re-enable black-listed NAT IPs. To direct BGP update messages to ACOS applications, enter the command at the BGP neighbor level.

Configuring NAT IP Black-holing via BGP

The following commands configure BGP configuration on ACOS on the receiver side:

```
ACOS(config)# router bgp 227
ACOS(config-router:)# bgp router-id 1.2.3.4
ACOS(config-router:)# neighbor 10.1.1.123 remote-as 123
ACOS(config-router:)# neighbor 10.1.1.123 acos-application-only [1]
ACOS(config-router:)# neighbor 10.1.1.123 route-map set_zone in [2]
ACOS(config-router:)# exit
ACOS(config)# route-map set_zone permit 100 [3]
ACOS(config-route-map)# set ddos zone ddos_zone [4]
```

The following command configures CGN to disable specific NAT IPs fallen into the configured BGP DDoS zone “ddos_zone”:

```
ACOS(config)# cgnv6 ddos-protection disable-nat-ip-by-bgp zone ddos_zone
```

NOTE:

- [1] The **acos-application-only** option must be configured. “neighbor 10.1.1.123 acos-application-only” must be configured on ACOS to treat the route update sent from 10.1.1.123 to be sent to ACOS applications. These routes are not installed in the routing table.
- [2] An in-bound route-map must be configured.
- [3] The **route-map set_zone** command associates the route to the “ddos zone ddos_zone”.
- [4] The **set ddos zone ddos_zone** command defines the zone name “ddos_zone”. This zone name must be identical to the zone name configured in the **cgnv6 ddos-protection disable-nat-ip-by-bgp zone ddos_zone** command.
- If the zone configured in CGN is incorrectly configured without matching the zone name configured in BGP, then CGN will miss the route updates from BGP. When the configuration is corrected, to retrieve the missed updates, use the **clear ip bgp** command on ACOS to refresh BGP entries. Then, CGN will receive all BGP updates, including the missed entries.

To display the list of NAT IPs disabled on BGP advertisement, use the **show ddos-protection disabled-ip-by-bgp** command:

```
ACOS# show cgnv6 ddos-protection disabled-ip-by-bgp
IP Address            NAT Pool Name
=================================
1.1.1.1                 2
1.1.1.2                 2
```

To clear the currently disabled NAT IP on BGP advertisement, use the following command:

```
clear cgnv6 ddos-protection disabled-ip-by-bgp {all | ip-address  ip-addr}
```

NOTE: If ACOS receives BGP advertisement for this NAT IP later, it will be disabled again.

Clear DDoS Entries

Use the **clear** commands to delete L3 and L4 DDoS entries. The **clear** command provides options to selectively remove some entries or all DDoS statistics can be cleared entirely.

L3 DDoS entries can be cleared based on a NAT IP netmask or NAT Pool. L4 DDoS entries can be cleared based on a NAT IP netmask, port, protocol, or based on NAT pool.

CLI Configuration

To clear NAT IP disabled by BGP advertisement, enter the following command at the global configuration level:

```
ACOS(config)# clear cgnv6 ddos-protection disabled-ip-by-bgp    {all | ip-address ipaddr netmask netmask}
```

To clear L3 DDoS entries, enter the following command at the global configuration level:

```
ACOS(config)# clear cgnv6 ddos-protection ip-entries    {all | ip-address ipaddr netmask netmask | nat-pool name}
```

To clear L4 DDoS entries, enter the following command at the global configuration level:

```
ACOS(config)# clear cgnv6 ddos-protection l4-entries    {all | address ipaddr netmask netmask | l4-proto num  | nat-pool name | port num}
```

For clearing L4 entries, a combination of NAT address, port, and protocol can be specified together, in any order. If one filter is already specified, the others are optional. Clearing L4 port entries will only clear TCP and UDP traffic for those ports.

To clear all DDoS statistics, enter the following command at the global configuration level:

```
ACOS(config)# clear cgnv6 ddos-protection statistics
```

Selective Filtering for LSN

As public addresses, NAT IPs are highly susceptible to malicious behavior, such as port scans and DDoS attacks. High volume attacks, such as NTP reflection or DNS response spoofing, often target a single destination port. While ACOS identifies and drops packets from such volumetric attacks, a high rate of attack traffic can affect CPU performance. To distribute the processing load and prevent a single CPU from being overwhelmed, ACOS provides an option to enable CPU round-robin, which balances attack traffic across all available CPUs.

ACOS supports selective filtering, which provides a DDoS protection mechanism for NAT pools and helps protect servers and CPUs from reflection and spoofing types of volumetric attacks. For selective filtering, ACOS tracks protocol packets per second rate limit. These limits are matched on a destination 2-tuple basis (NAT IP and NAT port). The thresholds are not configured for a specific destination 2-tuple. Rather, ACOS tracks the destination 2-tuple of all incoming packets and drops packets when the threshold is exceeded for any given destination 2-tuple.

On certain platforms, selective filtering can identify when packets are coming in at an abnormally fast rate. ACOS creates the entries in the logging table and drops the packets. The entries are of two types:

- L3 entry - Created when the destination IP is blocked
- L4 entry - Created when a destination IP and destination port combination is blocked

On supported platforms, ACOS supports selective filtering in hardware to avoid impacting the CPUs.

Software-Based Selective Filtering

When the traffic going to a specific destination IP and destination IP port exceed the configured thresholds, L4 entries are created in software tables. Future traffic matching those entries is dropped. These entries remain on the tables for 10 seconds (configurable value) and age out if their packets per second rate drops below the threshold. If the incoming traffic rate remains high, the table timeout refreshes.

On software, selective filtering thresholds are configurable for TCP or UDP traffic individually, other Layer 4 traffic as a group, or per source-IP address regardless of the packet protocol. While TCP and UDP can have separate rate limits, all other Layer 4 traffic is subject to the same limit on a per-protocol basis. However, one Layer 4 protocol being rate limited does not affect the other Layer 4 protocols. For example, the following command limits non-TCP and non-UDP Layer 4 traffic to 4000 packets per second:

```
ACOS(config)# cgnv6 ddos-protection packets-per-second other 4000
```

If over 4000 GRE packets match a given destination IP and destination IP port pair, then an L4 entry is created to drop GRE packets for that 2-tuple. Other Layer 4 protocols, such as MOBILE or AH, matching the same 2-tuple are still accepted, as long as they are under the 4000 packets per second rate limit. They cannot be configured individually so that the GRE limit is higher than the MOBILE or AH limit, for example.

LSN selective filtering is performed in two stages:

1. **Stage 1**: If the “bad” packets-per-second to a single NAT IP is greater than the configured DDoS protection packets-per-second IP threshold, then processing moves to stage 2.
2. **Stage 2**: Processing depends on the Layer 4 protocol:

- TCP/UDP – If the "bad" packets-per-second to a single (NAT IP:port) pair exceeds the configured threshold, then that pair gets the selective filtering entry. For example, if UDP packets that hit a NAT IP on port 5000 exceed the threshold, then only UDP packets to port 5000 will be blocked. Other UDP packets to that NAT IP will not be affected.
- Other Layer 4 protocols – If the "bad" packets-per-second to a single (NAT IP: Layer 4 protocol) pair exceeds the configured threshold, that pair gets an entry. For example, if GRE (ip protocol 47) packets to one NAT IP exceeds the threshold for Other protocols, then only GRE packets to that NAT IP will be blocked.

NOTE: Layer 4 packets-per-second is only checked when Layer 3 packets-per-second exceeds the configured minimum Layer 4 packets-per-second threshold. This ensures that unnecessary checks are avoided when total packets-per-second remains below the threshold.
For example, traffic to a specific NAT IP and port combination will be dropped if it exceeds 2000 packets-per-second. If the overall traffic rate to the NAT IP surpasses 10000 packets-per-second, the NAT IP will be quarantined, and all incoming traffic to it will be blocked.
It is important to ensure that the configured Layer 3 packets-per-second threshold is greater than the Layer 4 packets-per-second threshold. If the Layer 3 packets-per-second value is less than or equal to the Layer 4 packets-per-second value, the system will drop all traffic once the Layer 3 threshold is reached, and the Layer 4 packets-per-second check will never be triggered. This could lead to unintended behavior in DDoS scenarios.

Hardware-Based Selective Filtering

On supported platforms, destination 2-tuples (NAT IP and NAT port) can be programmed into the hardware. The hardware tables support rate limiting for both L3 and L4 drops; however, for L4 traffic, only TCP and UDP protocols are supported. This allows anomalous traffic to be dropped at the hardware level, reducing CPU load and improving protection against large-scale volumetric attacks.

NOTE:

- Entries are offloaded to the hardware only if the configured action is `drop`; for non-drop actions such as `log`, a log is generated but the packets are neither dropped in software nor offloaded to hardware.
- On platforms without special hardware capabilities, this feature is supported in software only.

Hardware offloading is supported on SPE Platforms (FTA). On these platforms, the Security and Policy Engine (SPE) stores the 2-tuple entries, and packets matching the L3 and L4 entries are dropped by the hardware. Like RAM, the SPE entries will be deleted if the ACOS device is reloaded or rebooted.

Configuring Selective Filtering for LSN

To configure selective filtering for LSN, perform the following steps:

1. Configure packet-per-second (PPS) rate limit for the desired protocol(s).
2. Enable DDoS protection if it is disabled.

NOTE: DDoS protection is enabled by default.

GUI Configuration

This feature currently is not supported in GUI.

CLI Configuration

To configure selective filtering for LSN, enter the following command at the global configuration level. The following example configures DDoS protection to selectively filter packets. The rate limits are increased to a limit of 5000 packets-per-second (PPS) for each IP address, 8000 PPS for TCP traffic, 6000 PPS for UDP traffic, and 50000 PPS for all other traffic. DDoS logging and protection are also enabled.

```
ACOS(config)# cgnv6 ddos-protection packets-per-second ip 5000 action drop
ACOS(config)# cgnv6 ddos-protection packets-per-second tcp 8000
ACOS(config)# cgnv6 ddos-protection packets-per-second udp 6000
ACOS(config)# cgnv6 ddos-protection packets-per-second other 50000
ACOS(config)# cgnv6 ddos-protection logging enable
ACOS(config)# cgnv6 ddos-protection enable
```

On the hardware, selective filtering is supported at the L3 level and at the L4 level for TCP and UDP traffic only. The default value per IP is 3000000, for TCP 3000, and for UDP 3000. The default value for all other Layer 4 protocols is 10000. The configurable rate limit for any protocol or per IP can range from 0 to 30000000 PPS.

NOTE: DDoS protection and logging are enabled by default. Use the **disable** option to disable them, if desired.

Viewing Selective Filtering Statistics

- To view selective filtering statistics, use the following show command:

```
ACOS# show cgnv6  ddos-protection statistics
L3 Entry Added                          0
L3 Entry Deleted                        0
L3 Entry Added to BGP                   0
L3 Entry Removed From BGP               0
L3 Entry Added to HW                    0
L3 Entry Removed From HW                0
Too Many L3 entries                     0
L3 Entry Match Drop                     0
HW L3 Entry Match Drop                  0
L4 Entry Added                          0
L4 Entry Deleted                        0
L4 Entry Added to HW                    0
L4 Entry Removed From HW                0
HW out of L4 Entries                    0
L4 Entry Match Drop                     0
HW L4 Entry Match Drop                  0
```

Table 11 : show DDoS-Protection statistics field descriptions

Field	Description
Entry Added	The number of destination NAT IP and destination NAT IP port pairs for a given protocol that were added as entries to the software table. Entries for TCP, UDP, per source IP, and separate Layer 4 protocols are counted separately
Entry Deleted	The number of destination NATP IP and destination NAT IP port pairs for a given protocol that were removed from the software table because their packets-per-second rates were under the threshold for 10+ seconds. Entries for TCP, UDP, per source IP, and separate Layer 4 protocols are counted separately.
Entry Added to HW	Software entries that were also added as a hardware entry on FTA supported platforms.
Entry Removed from HW	Entries deleted from the hardware on FTA supported platforms.

Table 11 : show DDoS-Protection statistics field descriptions

Field	Description
HW out of Entries	How many entries are not logged in the hardware due to limited space for programmed entries.
Entry Match Drop	How many packets are dropped at the software level because they matched an entry.
HW Entry Match Drop	How many packets are dropped by the FTA because they matched a hardware entry

- To view the selective filtering IP entries, use the following command:

```
ACOS(config)# show cgnv6 ddos-protection ip-entries
```

- To view the selective filtering L4 port entries, use the following command:

```
ACOS(config)# show cgnv6 ddos-protection l4-entries
```

For more information on the show commands, refer to the *Command Line Interface Reference Guide*.

Selective Filtering for Existing CGN Sessions

By default, selective filtering calculates the packets-per-second (PPS) rate based on packets received from the internet (outside) toward inside subscribers, i.e., packets destined for the NAT IP. However, in some cases, it is required to include packets from existing CGN sessions while computing the PPS rate.

To include existing session traffic in the PPS calculation, configure the following command at the global configuration level:

```
ACOS(config)# cgnv6 ddos-protection packets-per-second include-existing-session
```

The **include-existing-session** command is disabled by default.

NOTE: Selective filtering applies only to CGN sessions initiated after the feature is configured.

Selective Filtering Capacity – Hardware and Software Limits

Software Enforcement

When selective filtering is configured, some IP addresses can be blocked in software, and all packets destined for those addresses are dropped. The software can block up to 1024 L3 entries. There is no fixed limit for L4 entries as long as sufficient memory is available to store them.

Hardware Enforcement

On certain platforms, selective filtering entries can be offloaded to the hardware to reduce CPU load. However, if the hardware resources are exhausted, additional entries are automatically enforced in software, up to the software capacity limits.

The maximum number of hardware entries supported depends on SPE platforms is 256K (262,144). The hardware entries limit applies only when an IPv4 profile is configured. By default, IPv4 and IPv6 profiles are configured. Therefore, the maximum configurable entries supported are 128K.

Configurable Limits

If the provider's available NAT IP addresses reach their maximum and are all dropped at the hardware level during an attack, CGN service operation may be impacted. To prevent this, you can configure a lower limit for selective filtering entries in the hardware using the following command:

```
ACOS(config)# cgnv6 ddos-protection max-hw-entries <num>
```

NOTE: This command is only available on platforms that support selective filtering at the hardware-level.

When a lower limit is configured, additional entries cannot be added at the hardware or software level until the old entries age out (or are cleared). Removing the configured limit resets the limit to the default values.

IP Anomaly Filtering

This section describes various IP anomaly filtering techniques.

The following topics are covered:

IP Anomaly Filtering Based on Packet Deformities and Security Attacks

ACOS provides configurable protection against a range of IP packet anomalies. When IP anomaly filtering is enabled, ACOS checks inbound traffic for the specified anomalies and drops any packets that have the anomaly.

Table 12 lists the types of IP anomalies ACOS can detect and drop.

Table 12 : IP Anomalies ACOS Can Detect and Drop

Anomaly Class	**Network Layer**	
	Layer 3	**Layer 4**
Packet Deformities	Bad IP Header Len	TCP Bad Urgent Ofs
	Bad IP Flags	TCP Short Header
	Bad IP TTL	TCP Bad IP Length
	Oversize IP Payload	TCP Null Flags
	Bad IP Payload Len	TCP Null Scan
	Bad IP Fragment Ofs	TCP Syn & Fin
	Bad IP Checksum	TCP XMAS Flags
	Runt IP Header	TCP XMAS Scan
	IP-over-IP Tunnel Mismatch	TCP Syn Fragment
	VXLAN Tunnel Bad IP Length	TCP Fragmented Hdr
	NVGRE Tunnel Bad IP Length	TCP Bad Checksum
		TCP Option Error
		Runt TCP/UDP Header

Table 12 : IP Anomalies ACOS Can Detect and Drop

Anomaly Class	Network Layer	
	Layer 3	**Layer 4**
	IP-over-IP Tunnel Bad IP Length	UDP Short Header UDP Bad Length UDP Port Loopback UDP Bad Checksum
Security Attacks	LAND Attack Empty Fragment Micro Fragment IPv4 Options IP Fragment No IP Payload ICMP Ping of Death	TCP Null Flags TCP Null Scan TCP XMAS Flags TCP XMAS Scan TCP Syn & Fin TCP Syn Fragment TCP Fragmented Hdr UDP Kerberos Frag UDP Port Loopback

You can enable each of the groups of anomalies separately. For example, you can enable filtering and dropping of Layer 3 attacks independently of filtering and dropping of Layer 4 attacks.

Configuring IP Anomaly Filtering

Use the **`ip anomaly-drop`** to enable IP anomaly filtering.

To enable filtering for IP packets that exhibit predictable, well-defined anomalies, use the following command for different types of IP anomalies:

```
ACOS(config)# ip anomaly-drop packet-deformity layer-3
ACOS(config)# ip anomaly-drop packet-deformity layer-4
ACOS(config)# ip anomaly-drop security-attack layer-3
ACOS(config)# ip anomaly-drop security-attack layer-4
```

To display IP anomaly filtering statistics, use the following command:

```
ACOS(config)# show ip anomaly-drop statistics
IP Anomaly Statistics:
---------------------------
Land Attack Drop                          0
Empty Fragment Drop                       0
Micro Fragment Drop                       0
IPv4 Options Drop                         0
IP Fragment Drop                          0
Bad IP Header Len Drop                    0
Bad IP Flags Drop                         0
Bad IP TTL Drop                           11
No IP Payload drop                        0
Oversize IP Payload Drop                  0
Bad IP Payload Len Drop                   0
Bad IP Fragment Offset Drop               0
Bad IP Checksum Drop                      0
ICMP Ping of Death Drop                   0
TCP Bad Urgent Offset Drop                0
TCP Short Header Drop                     0
TCP Bad IP Length Drop                    0
TCP Null Flags Drop                       0
TCP Null Scan Drop                        0
TCP Syn and Fin Drop                      52
TCP XMAS Flags Drop                       0
TCP XMAS Scan Drop                        0
TCP Syn Fragment Drop                     0
TCP Fragmented Header Drop                0
TCP Bad Checksum Drop                     0
UDP Short Header Drop                     0
UDP Bad Length Drop                       0
UDP Kerberos Fragment Drop                0
UDP Port Loopback Drop                    0
UDP Bad Checksum Drop                     0
Runt IP Header Drop                       0
Runt TCP/UDP Header Drop                  0
IP-over-IP Tunnel Mismatch Drop           0
TCP Option Error Drop                     0
IP-over-IP Tunnel Bad IP Length Drop      0
VXLAN Tunnel Bad IP Length Drop           0
NVGRE Tunnel Bad IP Length Drop           0
```

NOTE: The counter for an anomaly increments only if filtering and dropping for that anomaly type is enabled.

IP Anomaly Filtering based on IPv6 Extension Headers

This section describes the IP anomaly filtering technique based on the IPv6 Extension Headers.

The following topics are covered in this section:

Overview

The IPv6 Extension Headers (EHs) contain supplementary information that helps routers and other network devices decide how to process and forward the IPv6 packets. However, these extension headers can be misused by attackers and have proved to be a security threat. The attackers can manipulate the options from Hop-by-Hop, Destination, and Routing extension headers and cause IP spoofing attacks. Remote attackers can also abuse these extension headers to craft special IPv6 packets that trigger DDoS attacks.

To handle the security threat presented by the IPv6 Extension Headers, you can configure the **`ip anomaly-drop ipv6-ext-header`** command. This command enables the detection and filtering of malformed IPv6 packets based on the extension header types.

NOTE:

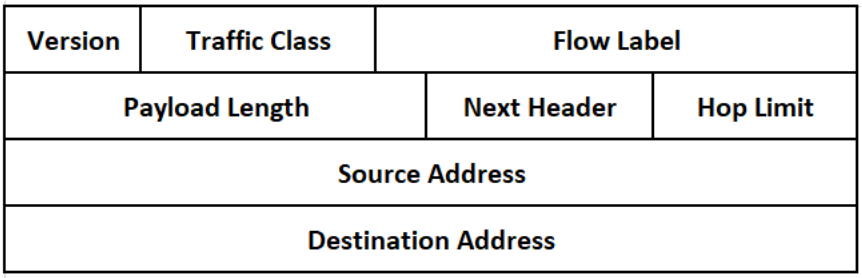
- This command can be enabled globally at the system level. It cannot be enabled at the partition level.
- Hardware FPGA does not accelerate the detection and filtering of IPv6 Extension Header based anomalies.

IPv6 Extension Headers

The IPv6 header has one Fixed Header (similar to the basic IPv4 header) and some (optional) Extension Headers. The Fixed Header has a fixed size (40 bytes) and has information essential for the routers to process and forward packets. The IPv6 Extension Headers (EHs) contain additional information that helps routers and other network devices determine how to process and forward the IPv6 packets.

The Figure 23 shows the structure of the IPv6 Fixed Header.

Figure 23 : IPv6 Fixed Header



The Fixed Header contains the following fields:

- Version - The version number of Internet Protocol (6 for IPv6).
- Traffic Class - The traffic class of the IPv6 packet.
- Flow Label - The flow label is assigned to the IPv6 packets that require special handling (by IPv6 routers).
- Payload Length - The total length of the payload.
- Next Header - This field indicates the type of header that immediately follows the Fixed Header. It may be an Extension Header or (in the absence of an Extension Header) the Upper Layer Protocol Data Unit (PDU).
- Hop Limit - The hop limit of the IPv6 packet. Every node that forwards the packet, decrements the value by one.
- Source Address - The source address of the IPv6 packet.
- Destination Address - The destination address of the IPv6 packet.

In IPv6, the additional or supplementary information is placed between the Fixed Header and the Upper Layer Header in the form of Extension Headers (EHs). Each EH is identified by a unique value. The following table contains the list of supported IPv6

EHs (supported as per RFC 2460 standards), the Option Type values and the recommended order of the EHs in an IPv6 packet.

Table 13 : Supported IPv6 Extension Headers

Order	Extension Header Type	Description	Next header code (Protocol Number)	Option Type values
1	Fixed IPv6 Header	40 bytes Fixed Header that contains information essentials for the routers.	-	-
2	Hop-by-Hop Options Header	Contains information to be processed by each router on the path. **NOTE**: If this header is present, it MUST be the first one that follows the Fixed Header.	0	All option types (0 -255)
3	Destination Options Header - 1 (first instance)	This header can appear twice in the IPv6 packet. The first instance contains information processed by the first and subsequent destination.	60	All option types (0-255)
4	Routing Options	This header contains information that supports making routing decision.	43	Routing type (0-255)
5	Fragment Header	This header contains information that supports communication	44	None

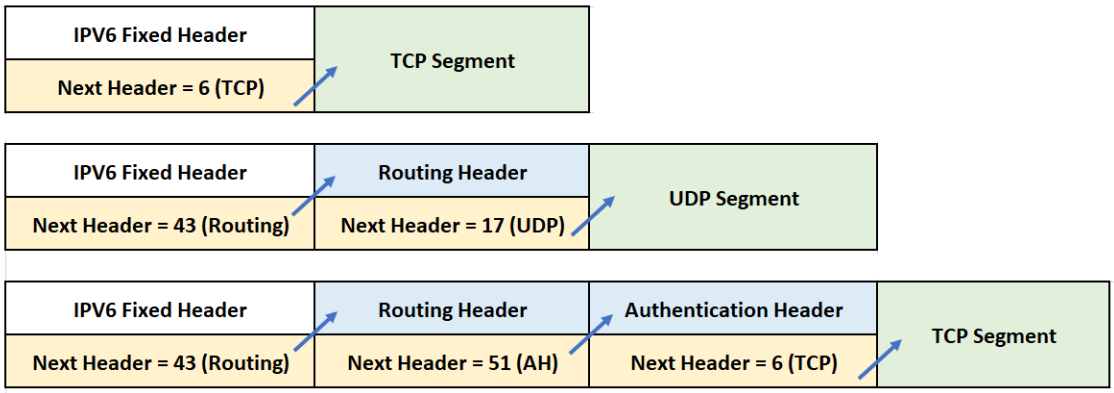
Order	Extension Header Type	Description	Next header code (Protocol Number)	Option Type values
		using fragmented packets.		
6	Authentication Header	This header contains information that supports integrity, authenticity and security of the IPv6 packets.	51	None
7	Encapsulation Security Payload Header	This header contains encryption information that provides security.	50	None
8	Destination Options Header - 2 (second instance)	This header can appear twice in the IPv6 packet. The second instance contains information processed by the final destination.	60	All option types (0-255)
9	Mobility Header	This header contains information for IPv6 mobility services.	135	None
-	No Next Header	This field indicated that there is no header after this header.	59	None

When EHs are present in a packet, the Next Header field of the Fixed Header points to the first EH that follows. If there is one more EH, then the first EH's Next-Header field points to the second one, and so on. The last EH’s Next-Header field points to the Upper Layer Header. Thus, all the headers point to the next header and are arranged in a chain-like manner.

NOTE: A Next Header field with a value 59 indicates that there are no headers after this header, not even the Upper-Layer Header.

The Figure 24 demonstrates the arrangement of EHs in a chain-like manner:

Figure 24 : Arrangement of Extension Headers



CLI Configuration

The **`ip anomaly-drop ipv6-ext-header`** command enables detection and filtering of anomalies based on IPv6 Extension Headers (EHs).

The following configurations are covered in this section:

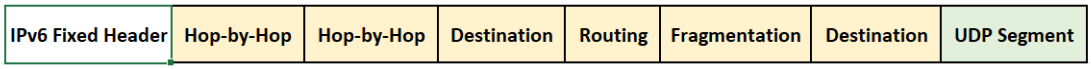
- Filtering Malformed IPv6 Packets
- Filtering Hop-by-Hop and Destination Extension Headers
- Filtering Routing Extension Headers
- Filtering Unknown Extension Headers
- Filtering Authentication Extension Header
- Filtering ESP Extension Header
- Filtering Fragmentation Extension Header
- Filtering Mobility Extension Header
- Viewing Anomaly Filtering Statistics

Filtering Malformed IPv6 Packets

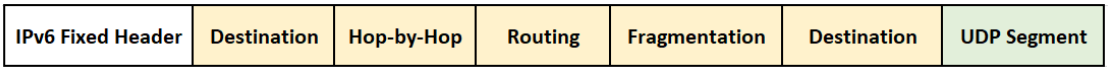
The RFC2460 standard recommends the order in which the EHs should be placed in an IPv6 packet (see Table 13). Additionally, the number of occurrences of the EHs is also mandated. All the EHs except the Destination Header can only appear once in the packet. The Destination Header can occur at most twice; once before the Routing Header and once before the Upper-Layer header. An IPv6 packet is considered as malformed if the extension header order is not followed or the number of occurrences of the extension header is different than expected.

Examples of Malformed IPv6 packets:

- **Example 1**: This packet is considered malformed because the Hop-by-Hop EH appears twice in the packet. As per the standard, only the Destination Header can appear twice (that too in a particular order).



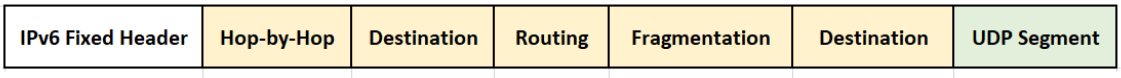
- **Example 2**: This packet is considered malformed because the EH order is incorrect. As per the standard, only the Hop-By-Hop EH can follow the Fixed Header; the Destination header cannot follow the Fixed Header.



To detect and filter such malformed IPv6 packets, configure the following command:

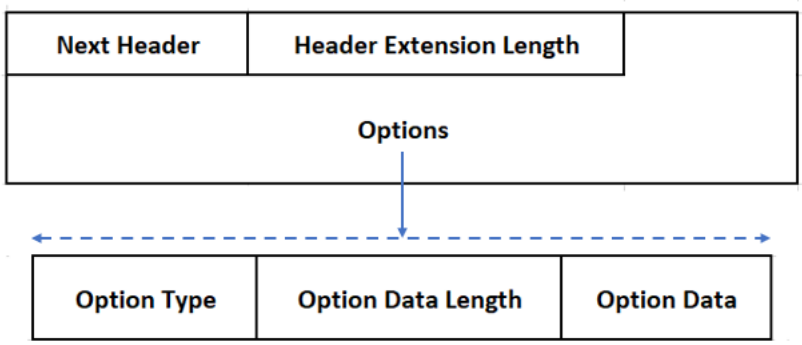
```
ACOS(config)# ip anomaly-drop ipv6-ext-header malformed
```

Consider the following IPv6 packet. This packet follows the correct order and number of occurrences of the EHs. Since the packets is not malformed, it will not be dropped if this command is configured.



Filtering Hop-by-Hop and Destination Extension Headers

The following figure shows the structure of the Hop-By-Hop and Destination Extension Header:



These headers contain the following fields:

- Next Header - Indicates the type of EH that immediately follows the Hop-by-Hop/Destination EH.
- Header Extension Length - Indicated the length of this EH (without including the length of the Next Header field).
- Options - This is a variable length field that contains one or more options. These options need to be processed by each router on the path. Each option further consists of the following fields:
  - Option Type - It is an 8 bit field that identifies the option type, which determines how the processing node handles the option.
  - Option Data Length - It is the length of the Option Data field (of this option).
  - Option Data - It is a variable length field that contains data specific to the option.

The Option Type field of the EH is frequently abused by attackers resulting in potential DDoS attacks. Such IPv6 packets with malicious Option Type fields can be filtered by configuring the **`ipv6-ext-header hop-by-hop/dest`** command. This command can be configured in the following ways:

- To filter packets with a specific Option Type:

  For Hop-By-Hop Header:

  ```
  ACOS(config)# ip anomaly-drop ipv6-ext-header hop-by-hop option-type <0-255>
  ```

  For Destination Header:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header dest option-type <0-255>
```

NOTE: Option Type 1 is often used for padding. As a result, this field is implicitly set in most IPv6 packets. If you configure the **option-type** as 1, most of the IPv6 packets, including the non-malicious packets, will be dropped

- To filter packets with a range of Option Types:

  For Hop-By-Hop Header:

  ```
  ACOS(config)# ip anomaly-drop ipv6-ext-header hop-by-hop option-type <0-255> to <0-255>
  ```

  For Destination Header:

  ```
  ACOS(config)# ip anomaly-drop ipv6-ext-header dest option-type <0-255> to <0-255>
  ```

- To filter packets containing Hop-By-Hop/Destination EH (irrespective of the Option Type):

  For Hop-By-Hop Header:

  ```
  ACOS(config)# ip anomaly-drop ipv6-ext-header hop-by-hop
  ```

  For Destination Header:

  ```
  ACOS(config)# ip anomaly-drop ipv6-ext-header dest
  ```

Configuration Examples:

- The following example enables filtering of packets that contain Destination EH with Option Type 110:

  ```
  ACOS(config)# ip anomaly-drop ipv6-ext-header dest option-type 110
  ```

- The following example enables filtering of packets that contain Hop-By-Hop EH header with Option Type in the range 194 to 201:

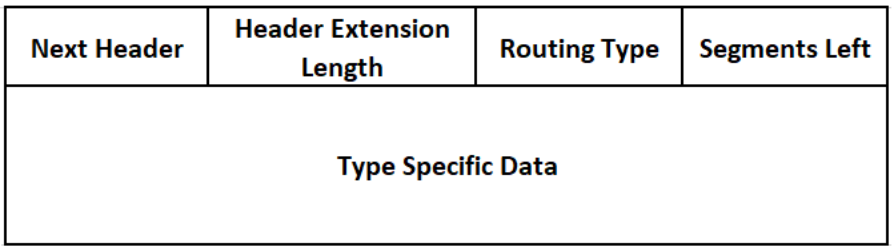
  ```
  ACOS(config)# ip anomaly-drop ipv6-ext-header hop-by-hop option-type 194 to 201
  ```

- The following example enables filtering of packets that contain Destination EH:

  ```
  ACOS(config)# ip anomaly-drop ipv6-ext-header dest
  ```

Filtering Routing Extension Headers

The following figure shows the structure of the Routing Extension Header:



The Routing EH contains the following fields:

- Next Header - Indicates the type of EH that immediately follows the Routing header.
- Header Extension Length - Indicated the length of this header (without including the length of the Next Header field).
- Routing Type - It is an 8 bit field that identifies the Routing Type.
- Segments Left - This field indicates the number of route segments yet to be traversed before reaching the final destination.
- Type Specific Data - It is a variable length field that contains data specific to the Routing Type.

Similar to the Hop-By-Hop and Destination Headers, the Routing Type field of the Routing EH is vulnerable to manipulations by attackers. The IPv6 packets with malicious Routing Type fields can be filtered by configuring the **`ipv6-ext-header routing`** command. This command can be configured in the following ways:

- To filter packets with a specific Routing Type:

  ```
  ACOS(config)# ip anomaly-drop ipv6-ext-header routing option-type <0-255>
  ```

  The following example enables filtering of packets that contain Routing header with Routing Type 110:

  ```
  ACOS(config)# ip anomaly-drop ipv6-ext-header routing option-type 110
  ```

- To filter packets with a range of Routing Types:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header routing option-type <0-255> to <0-255>
```

The following example enables filtering of packets that contain Routing header with Routing Type in the range 194 to 201:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header routing option-type 194 to 201
```

- To filter packets having Routing Extension header (irrespective of the Routing Type):

```
ACOS(config)# ip anomaly-drop ipv6-ext-header routing
```

The following example enables filtering of packets that contain Routing header:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header routing
```

Filtering Unknown Extension Headers

To filter packets that contain unknown extension header types, configure the following command:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header eh-type <0-255> [ to <0-255> ]
```

Using this command, you can drop a particular extension header type or a range of extension header types.

NOTE: You must specify values other than 0, 43, 44, 50, 51, 59, 60, and 135. These IDs are set for the eight well-known IPv6 extension headers

Configuration Examples:

- The following example enables filtering of packets that contain header type 110:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header eh-type 110
```

- The following example enables filtering of packets that contain the extension types in the range 194 to 201:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header eh-type 194 to 201
```

Filtering Authentication Extension Header

The Authentication header is used to provide origin authentication and security to the traffic flow. To filter packets containing the Authentication EH, configure the following command:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header auth
```

Filtering ESP Extension Header

The Encapsulation Security Payload (ESP) header contains encryption information, which is used to provide security, authentication, and confidentiality to the traffic flow. To filter packets that contain the ESP Extension Header, configure the following command:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header esp
```

Filtering Fragmentation Extension Header

The Fragment header is used by the IPv6 source to send large packets. To filter packets that contain the Fragmentation EH, configure the following command:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header frag
```

Filtering Mobility Extension Header

The Mobility header is used to support IPv6 Mobile services. To filter packets that contain the Mobility EH, configure the following command:

```
ACOS(config)# ip anomaly-drop ipv6-ext-header mobility
```

Viewing Anomaly Filtering Statistics

To view the packet filtering statistics based on IPv6 EHs, use the following command:

```
ACOS(config)# show ip anomaly-drop statistics
```

NOTE: The filtering of IPv6 Extension Header based anomalies is not logged.

For more information on command usage, refer to the *Command Line Interface Reference* guide.

Connection Rate Limiting

ACOS allows you to set a limit on the number of sessions allowed per source IP.

A common form of DDoS attacks are volumetric attacks, such as TCP SYN flooding. These attacks flood servers with a large number of packets, thereby consuming resources with open or half-open sessions. To mitigate these types of volumetric attacks, ACOS allows you to configure a connection limit. The connection rate limit allows you to set a maximum number of sessions allowed per source IP, thus preventing one user from consuming all of the server sessions and blocking legitimate traffic from other clients.

While volumetric DDoS attacks may slowly consume server resources over time, they may also happen within a short time frame. In the case where there is a connection limit per source IP, attacks may still attempt to bring down the server by using up the connection quota all at once. These attacks can be mitigated by configuring a connection rate limit.

For connection rate limiting, you configure a maximum number of connections a user can attempt to initiate per second. These limits are configurable on a per source IP basis. If the per second connection limit is exceeded, no new connections will be made, even if the number of total sessions per IP has not been exceeded.

This feature is applicable to CGN, NAT44, NAT64, DS-LITE, and 6rd-NAT64.

Configuring Connection Rate Limiting

To configure connection rate limiting, an LSN limit ID (LID) is created and the connection rate limit is configured. Then, the LSN LID is applied to a class list or a NAT pool as desired.

Configuring Connection Rate Limiting by Using the GUI

Perform the following steps in GUI:

1. Navigate to **CGN** > **LSN** > **LSN LID**.
2. Click **Create**.
3. Enter the **LID Number**.

4. Enter the **Connection Rate Limiting** value.
5. Enter or select other options. When finished, click **Create**.
6. Click **Save**.

Configuring Connection Rate Limiting by Using the CLI

The following example configures a LSN LID with a connection rate limit of 5000 connections per second.

1. First, an LSN NAT pool, which will be bound to the LSN LID, is configured.

```
ACOS(config)#cgnv6 nat pool LSN_POOL1 198.51.100.1 198.51.100.254
netmask /24
```

2. Create the LSN LID and configure the connection rate limit before binding the NAT pool.

```
ACOS(config)# cgnv6 lsn-lid 10
ACOS(config-lsn lid)# conn-rate-limit 5000
ACOS(config-lsn lid)# source-nat-pool LSN_POOL1
ACOS(config-lsn lid)# exit
```

3. Binds the LSN LID to a class-list.

```
ACOS(config)# class-list rate-limit
ACOS(config-class list)# 192.0.2.0/24 lsn-lid 10
```

To view the statistics for traffic exceeding the configured connection rate limit, use the following command:

```
ACOS# show cgnv6 lsn statistics
Traffic statistics for LSN:
---------------------------
...
Extended User-Quota Matched                25
Extended User-Quota Exceeded               13
Data Session User-Quota Exceeded           67
Conn Rate User-Quota Exceeded              15
TCP Full-cone Session Created              14
TCP Full-cone Session Freed                29
UDP Full-cone Session Created              6
...
```

The “Conn Rate User-Quota Exceeded” statistic reflects how many connections, past the quota limit, have been attempted from all source-IPs for which connection rate limiting is configured. It is a comprehensive statistic for the ACOS device.

Reduced CPU Overhead for CPU Round Robin

When CPU Round Robin is triggered, the packets are distributed across all available CPUs for processing in order to avoid oversubscribing on the targeted CPU. In some cases though, throughput and CPU utilization remained high until the excessive traffic ended. To reduce the overhead, the ACOS code path drops SYN packets earlier on in order to decrease use of the targeted CPU.

On the targeted CPU, incoming SYN packets are checked whether or not they are IPv4 packets, and if so, whether or not the destination IP is a NAT IP address. If those conditions are not met, the packets are dropped. If those conditions are met, then ACOS checks if the packet is valid for establishing a session. Invalid packets are dropped, while valid packets create a new session.

If LSN is enabled, then ACOS checks for an existing full cone session, port reservation, or ALG session. If Fixed NAT is enabled, then ACOS checks for an existing full cone session or ALG session. If none of those conditions are met, then the packet is dropped.

CPU round robin for CGN is enabled by default. All dropped packets increment the “L4 Out-of-State packets” in the **`show cgnv6 l4 debug`** command.

Use the **`show cgnv6 ddos-protection l4-entries`** command to view L4 entries.

Use the **`show cgnv6 ddos-protection ip-entries`** command to view all IP entries, including normal entries.

SYN Cookie

This chapter describes the SYN cookie feature and how it helps protect ACOS devices against disruptive SYN-based flood attacks.

The following topics are covered:

Overview of SYN Cookie

SYN cookie protects against TCP SYN flood attacks. When SYN cookie is enabled, the ACOS device can continue to serve legitimate clients during these attacks, while preventing illegitimate traffic from consuming system resources.

The SYN cookie is required for detecting SYN flooding from inside client. The existing DDoS detects the SYN flooding from outside client. Hairpin is not supported, so the existing DDoS can be used.

SYN Cookie works for LSN/DS-lite/NAT64/6rd-NAT64/Fixed-NAT, one-to-one NAT, and for all ALGs.

The session sync happens only after the connection is fully established. The SYN Cookie connection setup cannot be done across devices.

NOTE: Hardware SYN cookie is not supported.

The following topics are covered in this section:

SYN Flood Attacks

During a TCP SYN flood attack, an attacker sends many TCP SYN Requests to a network device, such as a server. The server replies with a standard SYN-ACK message. However, rather than reply to this attempt at establishing a 3-way handshake with the standard ACK, an attacker ignores the reply and creates a “half-open” TCP connection. System resources are consumed because the device waits for a response from the client that never arrives.

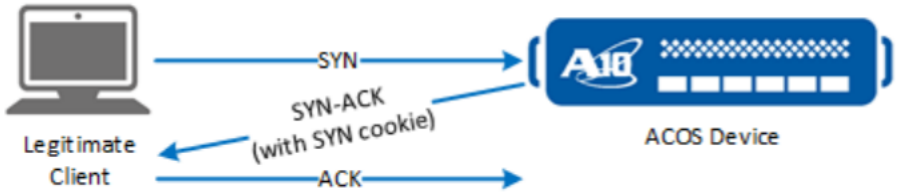
Under large-scale attacks, excessive half-open connections cause a network device’s TCP connection queue to become full. This over-subscription prevents the device from establishing new connections with legitimate clients.

Identifying SYN Flood Attacks

The graphics in this section illustrate how the ACOS device determines whether a particular TCP connection is from a legitimate request or if it is part of a SYN flood attack.

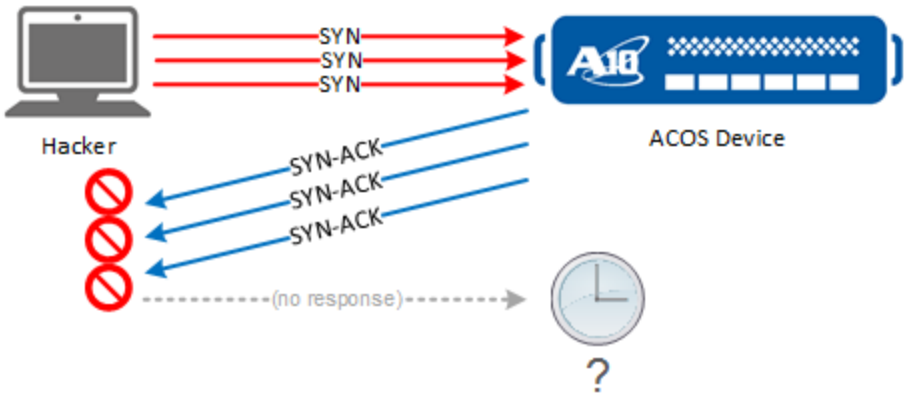
Figure 25 depicts a typical 3-way TCP handshake, which includes a SYN request from the client, the SYN-ACK reply from the ACOS device, and finally, an ACK from the client to the ACOS device.

Figure 25 : SYN-ACK Handshake (Legitimate Client)



However, SYN flood attacks (Figure 26) can cripple a network by sending multiple SYN requests to a network device. The device responds to these SYN requests with SYN-ACKs and waits for responses from the client that never arrive. These bogus requests create many “half-open” sessions, which wastes system memory and other system resources. The state of being oversubscribed reduces the device’s free resources, which prevents it from accepting requests from legitimate clients.

Figure 26 : SYN-ACK Handshake (Hacker)



Enabling SYN cookie mitigates the damage caused by such DoS attacks by preventing the attacks from consuming system resources.

TCP connections for which the ACOS device did not receive an ACK from the client is identified as belonging to a SYN flood attack, and this information is displayed with the counter in the output of the show command.

ACOS SYN cookie Protection

By enabling SYN cookie, the ACOS device's TCP connection queue is prevented from filling up during TCP SYN flood attacks. When a client sends an SYN request, the ACOS device responds with a SYN cookie. This response is a special type of SYN ACK message.

SYN cookie prevents hackers from consuming excessive system resources by encoding the necessary state information for the client connection in a TCP sequence number. Rather than storing state information for each TCP session, the sequence number in the SYN cookie acts as a shorthand, which allows the ACOS device to compress much of the session information into a smaller amount of data.

This sequence number is sent to the client as a SYN-ACK packet. When a legitimate client receives this information, it replies with an ACK that contains the sequence number plus 1.

When the SYN ACK that contains the sequence number from the client is received, the ACOS device reconstructs the connection information and establishes a connection with that client.

If the SYN Request is part of an attack, the attacker does not send an ACK to the ACOS device. The ACOS device sends a SYN cookie, but the attacker does not receive it (or may choose to ignore it), and the ACOS device does not establish a connection.

Dynamic SYN Cookie

The SYN Cookie can be triggered by number of half-open TCP sessions. You can configure on-threshold and on-timeout for SYN cookie. When there are no TCP SYN attacks, the TCP options are preserved.

The following CLI is provided to configure SYN cookie:

```
ACOS(config)#cgnv6 ddos-protection syn-cookie enable [tcp-half-open on-threshold on-limit [on-timeout timeout-value]]
```

You can configure the following dynamic SYN cookie options:

- On-threshold – specifies the maximum number of concurrent half-open TCP connections that are allowed on the ACOS device, before SYN cookie is enabled. If the number of half-open TCP connections exceeds the on-threshold value, the ACOS device enables SYN cookie.
- On-timeout – specifies the time how long the SYN cookie will be enabled. The default time is 120 seconds.

Example

Configure tcp syn cookie which is triggered by tcp-half-open and disable after two minutes.

```
cgnv6 ddos-protection syn-cookie enable tcp-half-open on-threshold 1000
on-timeout 120
```

Configuring SYN Cookie

During the SYN Cookie configuration if the server does not respond within the specified timeout duration, the ACOS resends the SYN packet to outside server when it detects the retransmit of the first data segment. After the session timeout, the data segments sent by the client can trigger RST if `cgnv6 lsn reset-on-error` is configured.

The following sections describe how to enable SYN cookie support and configure advanced features.

The following sections are covered:

- Enabling SYN Cookie Support
- Enabling SYN Cookie in CGN + Firewall Setup
- Modifying the Threshold for TCP Handshake Completion
- Viewing SYN Cookie Statistics

Enabling SYN Cookie Support

SYN cookie is supported on all ACOS devices.

To enable software-based SYN cookie, use the **`syn-cookie`** command. For example:

```
ACOS(config)#cgnv6 ddos-protection syn-cookie enable [tcp-half-open on-threshold on-limit [on-time timeout-value]]
```

This command configures TCP SYN cookie protection for CGN. It is triggered after the value in tcp-half-open sessions exceeds the configured threshold and is disabled after the timeout value expires.

SYN Cookie in CGN + Firewall Deployment

SYN cookie protection can also be applied per firewall rule. If SYN cookie is applied at the global CGN level as well as in a firewall rule that also permits cgnv6 traffic, the configuration in the firewall rule overrides the global CGN configuration.

The following example configures SYN cookie protection for CGN and in a firewall rule that applies CGN policy to the rule.

```
ACOS(config)#cgnv6 ddos-protection syn-cookie enable tcp-half-open on-threshold 100 on-timeout 120
!
ACOS(config)#rule-set rs1
ACOS(config-rule set:rs1)#rule 3
ACOS(config-rule set:rs1-rule:3)#action-group
ACOS(config-rule set:rs1-rule:3-action-group)#permit cgnv6 lsn-lid 1
ACOS(config-rule set:rs1-rule:3-action-group)#permit tcp syn-cookie enable
```

In the above example, the traffic matching rule will always have SYN cookie protection enabled, even though at the global CGN level, a threshold and timeout are specified. For details on configuring SYN cookie for Firewall rule, see Firewall Configuration Guide and Command Line Interface Reference.

Modifying the Threshold for TCP Handshake Completion

To modify the threshold for TCP handshake completion, use the **ip tcp syn-cookie threshold** global configuration command.

For example, to set the threshold to 3 seconds:

```
ACOS(config)# ip tcp syn-cookie threshold 3
```

Viewing SYN Cookie Statistics

This section describes how to view SYN cookie statistics by using the CLI.

The following command displays SYN cookie statistics:

```
ACOS-131-Active(config)#show cgnv6 ddos-protection statistics
L3 Entry Added                                  0
L3 Entry Deleted                                0
L3 Entry Added to BGP                           0
L3 Entry Removed From BGP                       0
L3 Entry Added to HW                            0
L3 Entry Removed From HW                        0
Too Many L3 entries                             0
L3 Entry Match Drop                             0
HW L3 Entry Match Drop                          0
L3 Entry Drop due to HW Limit Exceeded          0
L4 Entry Added                                  0
L4 Entry Deleted                                0
L4 Entry Added to HW                            0
L4 Entry Removed From HW                        0
HW out of L4 Entries                            0
L4 Entry Match Drop                             0
HW L4 Entry Match Drop                          0
L4 Entry Drop due to HW Limit Exceeded          0
TCP SYN cookie SYN ACK Sent                     7
TCP SYN cookie verification passed              7
TCP SYN cookie verification failed              0
```

Table 14 displays the fields that appear in the CLI output of the **`show cgnv6 ddos-protection statistics`** command.

Table 14 : show ddos-protection statistics fields

Field	Description
TCP SYN cookie SYN ACK Sent	The number of TCP SYN cookie sent.
TCP SYN cookie verification passed	The number of TCP SYN cookie for which the responding ACK passed the SYN cookie check.
TCP SYN cookie verification failed	The number of TCP SYN cookie for which the responding ACK failed the SYN cookie check.

Enabling Logging for DDoS Protection

Event logging for DDoS protection must be enabled in order to log and view the blacklisted NAT IP addresses. Even if the configured action is to log an event when a NAT IP is under a DDoS attack, the event will not be logged if DDoS protection logging is disabled.

When enabling the DDoS protection logging, you can choose to send the logs to the logging servers as follows:

- Local—Logs are sent to the local buffer and can be viewed using the **`show log`** command.
- Remote—Logs are sent to the remote syslog server and IPFIX collectors.
- Both—Logs are sent to both local buffer and remote servers.

The log protocols supported are Syslog and NetFlow. For Syslog, both CEF and ASCII formats are supported.

To enable event logging for DDoS protection, enter the following command at the global configuration level and configure one of the options:

```
ACOS(config)# cgnv6  ddos-protection logging enable
ACOS(config)# cgnv6  ddos-protection logging enable [local | remote |
both]
```

To disable event logging for DDoS protection, enter the following command at the global configuration level:

```
ACOS(config)# cgnv6  ddos-protection logging disable
```

To display the CGNAT logging show counters, see *Command Line Reference*.

Enhanced User Visibility

This chapter describes the enhanced user visibility feature that can be configured on LSN and NAT64.

The following topics are covered:

Overview

The enhanced user visibility feature lets you track the peak session utilization, NAT port utilization, and aggregated upstream and downstream byte and packet count per subscriber.

The subscriber information gathered from enhanced user visibility helps to detect anomaly in the subscriber behavior. The information can be used for allocating user-quota values for sessions and ports and for provisioning NAT IPs appropriately based on the subscriber usage.

By default, the enhanced user tracking is not enabled. It can be configured using the `cgnv6 lsn enhanced-user-tracking` command in the configuration mode.

If the `enhanced-user-tracking` command is enabled, a log message is generated by default every time a session is created or deleted for a user. For more information about enhanced-user-tracking logs, see *Traffic Logging Guide for IPv6 Migration*.

The enhanced user visibility feature is supported for LSN and NAT64.

Enabling Enhanced User Tracking

The enhanced user tracking feature must be configured at the partition level.

Using the CLI

Use the following command to enable the log using the CLI:

```
ACOS(config)# cgnv6 lsn enhanced-user-tracking
```

Using the GUI

To enable the enhanced user tracking log for LSN and NAT64 using the GUI:

1. Go to **CGN > LSN > Global**.
2. On the Update LSN Global page, under General Fields, select the check box for **Enhanced User Tracking**.

3. To view the subscriber information for LSN, go to **CGN > LSN > Stats > Subscriber Information**.
4. To view the subscriber information for NAT64, go to **CGN > NAT64 > Stats > Subscriber Information**.

Displaying the Enhanced User Tracking Information

To display the enhanced user tracking information for LSN, use the following CLI command:

```
ACOS(config)# show cgnv6 lsn enhanced-user-tracking
LSN Enhanced User Tracking:
Inside Addr   NAT Addr       TCP Current   UDP Current   ICMP Current   Session
Current TCP Peak   UDP Peak   ICMP Peak   Session Peak   Lifetime Sessions
Upload Packets   Upload load Packets   Download Bytes   NAT Pool Name
--------------------------------------------------------------------------
--------------------------------------------------------------------------
-----------------------------------------------------
3.3.3.91      15.15.15.120   0             64            0              24
         64         0           24              24                  0
   0                     0                lsn
LSN Enhanced User Tracking Count: 1
```

Table 15 describes the statistics displayed in the output:

Table 15 : LSN Enhanced User Tracking Statistics

Field	Description
Inside Address	The private, internal IP address.
NAT Address	The public IP address that is mapped to the internal IP address.
TCP Current	The current number of TCP connections per user.
UDP Current	The current number of UDP connections per user.
ICMP Current	The current number of ICMP connections per user.
Session Current	The current number of connections per session per user.
TCP Peak	The peak number of TCP connections per user.

Table 15 : LSN Enhanced User Tracking Statistics

Field	Description
UDP Peak	The peak number of UDP connections per user.
ICMP Peak	The peak number of ICMP connections per user.
Session Peak	The peak number of connections per session per user.
Lifetime Sessions	The total number of lifetime sessions.
Upload Packets	The total number of upstream packet count per session.
Upload Bytes	The total number of upstream byte count per session.
Download Packets	The total number of downstream packet count per session.
Download Bytes	The total number of downstream byte count per session.
NAT Pool Name	The NAT pool to which the IP address belongs.

To display the enhanced user tracking information for NAT64, use the following CLI command:

```
ACOS(config)#show cgnv6 nat64 enhanced-user-tracking
NAT64 Enhanced User Tracking:
Inside IPv6  Prefix  NAT Address  TCP Current  UDP Current  ICMP Current
Session Current  TCP Peak  UDP Peak  ICMP Peak  Session Peak  Lifetime
Sessions  Upload Packets  Upload Bytes  Download Packets  Download Bytes
NAT Pool Name
--------------------------------------------------------------------------
--------------------------------------------------------------------------
--------------------------------------------------------------------------
----------
3001          64         -               0             0            1
  1               1           0           0             0                   0
           0                      0          0             0
nat64
NAT64 Enhanced User Tracking Count:1
```

Table 16 describes the statistics displayed in the output:

Table 16 : NAT64 Enhanced User Tracking Statistics

Field	Description
Inside IPv6	The private, internal IPV6 address.
Prefix	The user quota prefix configured for specified addresses.
NAT Address	The public IP address that is mapped to the internal IP address.
TCP Current	The current number of TCP connections per user.
UDP Current	The current number of UDP connections per user.
ICMP Current	The current number of ICMP connections per user.
Session Current	The current number of connections per session per user.
TCP Peak	The peak number of TCP connections per user.
UDP Peak	The peak number of UDP connections per user.
ICMP Peak	The peak number of ICMP connections per user.
Session Peak	The peak number of connections per session per user.
Lifetime Sessions	The total number of lifetime sessions.
Upload Packets	The total number of upstream packet count per session.
Upload Bytes	The total number of upstream byte count per session.
Download Packets	The total number of downstream packet count per session.
Download Bytes	The total number of downstream byte count per session.
NAT Pool Name	The NAT pool to which the IP address belongs.

User Quotas Based on IPv6 Prefix

This chapter provides information about how to base user quota configuration on an IPv6 prefix for simplifying deployment.

The following topics are covered:

Overview

This capability enables you apply a user quota prefix length over an entire subnet, extending NAT64 and DS-Lite user quota control over a collection of addresses, instead of a single source IP address.

Consider the following information:

- You can apply a user quota prefix length on a global level or per LSN LID basis. The user quota prefix length set for an LSN LID overrides the global configuration value. If the user quota prefix is not configured at the LSN LID level, the global configuration will be used.
- When the user-quota-prefix-length is configured, the user-quota udp/tcp/icmp/session configured in lsn-lid will be applied to each prefix-based user.
- The udp/tcp/icmp/session count of a prefix-based user is the sum of udp/tcp/icmp/session count of all clients using this prefix.
- The user-quota-prefix-length must be greater or equal to the ipv6-prefix-length. For more information about ipv6-prefix-length, see *Command Line Reference Guide* and *Scaleout Configuration Guide*.
- This feature applies to all 64-bit platforms.
- For the command **`show cgnv6 nat64 user-quota-sessions`**, if a user quota prefix length is configured, only the prefix quota is displayed. If the prefix quota is not set, only the user quota session is displayed.

Configuring User Quota Prefix Length

Configuring User Quota Prefix Using the GUI

You can configure the user quota prefix length by using the GUI.

Configuring User Quota Prefix Length Globally

1. Navigate to **CGN > NAT64**.
2. In User Quota Prefix Length, enter a value between 1-128.

   The default prefix length is 128.
3. Click **Update**.

Configuring User Quota Prefix Length Per LSN LID

1. Navigate to **CGN > LSN > LSN LID**.
2. Click **Create** or **Edit**.
3. In **User Quota Prefix Length**, enter a value between 1-128.
4. Click **OK**.

Configuring User Quota Prefix Length Using the CLI

You can configure the user quota prefix length by using the CLI.

Configuring User Quota Prefix Length Globally

Enter the following command to configure a global prefix length for NAT64:

```
ACOS(config)# cgnv6  nat64 user-quota-prefix-length  22
```

Enter the following command to configure a global prefix length for DS-Lite:

```
ACOS(config)# cgnv6  ds-lite user-quota-prefix-length  22
```

You can select a value between 1-128. By default, the global user quota prefix length for is 128.

Configuring User Quota Prefix Length Per LSN LID

Enter the following commands to configure a prefix length per LSN LID:

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn lid)# user-quota-prefix-length 96
```

You can select a value between 1-128. By default, the LSN LID user quota prefix length is set to the global value.

CLI Example 1

The following example applies a user quota prefix to multiple LSN LIDs.

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn lid)# user-quota-prefix-length 96
ACOS(config-lsn lid)# cgnv6 lsn-lid 2
ACOS(config-lsn lid)# user-quota-prefix-length 73
```

CLI Example 2

This example configures user quota with multiple NAT64 prefix lengths.

```
ACOS(config)# class-list 2
ACOS(config-class list)# 2001:55:1:1::/64 lsn-lid 1
ACOS(config-class list)# 2001:55:2:2::/64 lsn-lid 2
ACOS(config-class list)# 2001:55:3:3::/96 lsn-lid 3
ACOS(config-class list)# exit
ACOS(config)# cgnv6 nat64 prefix 2003::/96
ACOS(config)# cgnv6 nat64 prefix 2008:88::/96 class-list 2
ACOS(config)# cgnv6 nat64 user-quota-prefix-length 96
ACOS(config)# cgnv6 nat64 inside source class-list 2
```

CLI Example 3

The following example uses the class list acts as the classifier:

```
ACOS(config)# class-list ipv6
ACOS(config-class list)# 3001::/64 lsn-lid 1
ACOS(config-class list)# exit
ACOS(config)# class-list lsn
ACOS(config-class list)# 1.1.1.1/32 lsn-lid 1
ACOS(config-class list)# exit
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn-lid)# user-quota-prefix-length 64
ACOS(config-lsn-lid)# user-quota session 1000
ACOS(config-lsn-lid))# exit
ACOS(config)# cgnv6 lsn inside source class-list lsn
ACOS(config)# cgnv6 nat64 inside source class-list ipv6
```

NOTE: The following is a list of configuration notes for this CLI example:
- The class-list is bound at the partition level.
- The LID is bound to the subnets under the class-list.
- The LID specifies the user-quota/prefix quota.
- Prefix quota applies to IPv6 addresses.

Displaying User Quota Session Information

To display user quota session information, use the **`show cgnv6 nat64 user-quota-sessions`** command for NAT64 sessions and the **`show cgnv6 ds-lite user-quota-sessions`** command for DS-Lite sessions.

NOTE: When the user-quota-prefix-length is configured, the show command output displays only prefix-based users.

In the following show command output, the prefix length is set to 96 and a connection is made with the address 2001::100.

```
ACOS# show cgnv6 nat64 user-quota-sessions
NAT64 User-Quota Sessions:
Inside IPv6   Prefix  NAT Address      ICMP   UDP    TCP    Session  Pool
  LID  Flag
------------------------------------------------------------------------
-------------
2001::          96    8.8.8.143        0      0      1      1        test2
  2     -
```

In this display, the prefix length is changed to 64. The new user quota is applied to a connection made through the address 2001::101, while the first user continues to use the 96 user quota. If the previous user disconnects and then returns, the 64 user quota will point to this user.

```
ACOS# show cgnv6 nat64 user-quota-sessions
NAT64 User-Quota Sessions:
Inside IPv6   Prefix  NAT Address      ICMP   UDP    TCP    Session  Pool
  LID  Flag
------------------------------------------------------------------------
-------------
```

```
2001::          64     8.8.8.102         0       0       1       1        test2
 2      -
2001::          96     8.8.8.143         0       0       2       2        test2
 2      -
Total User-Quota Sessions Shown: 2
```

This example shows when connections from 2001::100 are complete and the previous user quota pointed to this address is removed.

```
ACOS# show cgnv6 nat64 user-quota-sessions
NAT64 User-Quota Sessions:
Inside IPv6    Prefix NAT Address       ICMP    UDP     TCP     Session  Pool
  LID   Flag
--------------------------------------------------------------------------------
--------------
2001::          64    8.8.8.102         0       0       1       1        test2
2       -
Total User-Quota Sessions Shown: 1
```

When a user with address 2001::100 reconnects, the new user quota is applied.

```
NAT64 User-Quota Sessions:
Inside IPv6    Prefix NAT Address       ICMP    UDP     TCP     Session  Pool
   LID  Flag
--------------------------------------------------------------------------------
--------------
2001::            64    8.8.8.102         0       0       1       1        test2
  2      -
2001::            64    8.8.8.143         0       0       2       2        test2
  2      -
Total User-Quota Sessions Shown: 2
```

TCP Proxy on CGN/IPv6 Platform

This chapter explains how to enable the use of a TCP-proxy virtual port using CGN pool instead of the regular SLB NAT pool.

The following topics are covered:

Overview

- The feature takes advantage of TCP proxy in ADC/SLB using CGN pool instead of the regular SLB NAT pool.
- The **allow-slb-cfg enable** command allows SLB objects to be configured on a CGN partition This CLI command is not supported for any other SLB features. TCP-proxy virtual port must be configured along with a wildcard IPv4/IPv6 virtual server. This CLI command is used for supporting IP address insertion in HTTPS requests and for TCP-proxy.
- While configuring the virtual server, use the **use-cgnv6** sub-option as a source NAT option to configure a TCP-proxy virtual port.
- This feature supports only NAT64, CGN/NAT44, Fixed-NAT NAT44, and Fixed-NAT NAT64. All other CGN IPv6Migration technologies are not supported.
- The destination-based rule-list bound to Fixed-NAT configuration is supported for TCP-proxy.
- This feature supports CGN logging using CGN pool instead of the regular SLB NAT pool. The logging type supported is Syslog only. For more information about CGN logging, see *Traffic Logging Guide for IPv6 Migration.*

Configuring TCP Proxy

1. The following command enables the configuration of SLB objects in CGN partition:

```
ACOS(config)# allow-slb-cfg enable
```

NOTE: Prior to configuring any SLB objects, this command must be used in order to allow SLB objects to be configured on a CGN partition.

2. The following commands configure class-lists for clients:

```
ACOS(config)# class-list lsn
ACOS(config-class list)# 25.25.25.0/24 lsn-lid 1
```

```
ACOS(config)# class-list nat64
ACOS(config-class list)# 2001:db8::/64 lsn-lid 1
```

3. The following commands configure virtual LANs:

```
ACOS(config)# vlan 118
ACOS(config-vlan:118)# tagged ethernet 1
ACOS(config-vlan:118)# router-interface ve 118

ACOS(config)# vlan 119
ACOS(config-vlan:119)# tagged ethernet 3
ACOS(config-vlan:119)# router-interface ve 119
```

4. The following commands create the Ethernet interfaces connected to the firewalls:

```
ACOS(config)# interface ethernet 1
ACOS(config-if:ethernet:1)# enable
ACOS(config-if:ethernet:1)# exit

ACOS(config)# interface ethernet 2
ACOS(config-if:ethernet:2)# enable
ACOS(config-if:ethernet:2)# exit

ACOS(config)# interface ethernet 3
ACOS(config-if:ethernet:3)# enable
ACOS(config-if:ethernet:3)# exit
```

5. The following commands create the real servers or clients, and then enable promiscuous mode:

```
ACOS(config)# interface ve 118
ACOS(config-if:ve:118)# ip address 25.25.25.1 255.255.255.0
ACOS(config-if:ve:118)# ip allow-promiscuous-vip
ACOS(config-if:ve:118)# ip nat inside
ACOS(config-if:ve:118)# ipv6 address 2001:db8::2:15/96
ACOS(config-if:ve:118)# ipv6 nat inside
ACOS(config-if:ve:118)# exit

ACOS(config)# interface ve 119
ACOS(config-if:ve:119)# ip address 30.30.30.1 255.255.255.0
ACOS(config-if:ve:119)# ip nat outside
```

```
ACOS(config-if:ve:119)# ipv6 address a:b::c:f/96
ACOS(config-if:ve:119)# ipv6 nat outside
ACOS(config-if:ve:119)# exit
```

6. The following command configures a pool of IP addresses for use by source NAT:

```
ACOS(config)# ip nat pool p2 30.30.30.78 30.30.30.78 netmask /24
```

7. The following commands configure a real server:

```
ACOS(config)# slb server s 30.30.30.30
ACOS(config-real server)# port 80 tcp
ACOS(config-real server-node port)# exit
ACOS(config-real server)# exit

ACOS(config)# slb server s1 a:b::c:d
ACOS(config-real server)# port 80 tcp
ACOS(config-real server-node port)# exit
ACOS(config-real server)# exit
```

8. The following commands configures the service group:

```
ACOS(config)# slb service-group sg tcp
ACOS(config-slb svc group)# health-check-disable
ACOS(config-slb svc group)# member s 80
ACOS(config-slb svc group-member:80)# exit
ACOS(config-slb svc group)# exit

ACOS(config)# slb service-group sg1 tcp
ACOS(config-slb svc group)# member s1 80
ACOS(config-slb svc group-member:80)# exit
ACOS(config-slb svc group)# exit
```

9. The following commands configure the virtual server:

```
ACOS(config)# slb virtual-server vs 0.0.0.0
ACOS(config-slb vserver)# port 0 tcp-proxy
ACOS(config-slb vserver-vport)# source-nat use-cgnv6
ACOS(config-slb vserver-vport)# service-group sg
ACOS(config-slb vserver-vport)# no-dest-nat
ACOS(config-slb vserver-vport)# exit
ACOS(config-slb vserver)# exit
```

```
ACOS(config)# slb virtual-server vs1 ::
ACOS(config-slb vserver)# port 0 tcp-proxy
ACOS(config-slb vserver-vport)# source-nat use-cgnv6
ACOS(config-slb vserver-vport)# service-group sg
ACOS(config-slb vserver-vport)# exit
ACOS(config-slb vserver)# exit
```

10. The following command binds the class list to the LSN feature:

```
ACOS(config)# cgnv6 lsn inside source class-list lsn
```

11. The following command configures CGN pools:

```
ACOS(config)# cgnv6 nat pool p1 30.30.30.79 30.30.30.79 netmask /24
```

12. The following commands configure a LSN rule-list:

```
ACOS(config)# cgnv6 lsn-rule-list s
ACOS(config-lsn-rule-list)# ip 30.30.30.30/32
```

13. The following commands configure a LSN_LID and add the pool to it:

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn-lid)# source-nat-pool p1
ACOS(config)# exit
```

14. The following command binds the class list to the NAT64 feature:

```
ACOS(config)# cgnv6 nat64 inside source class-list nat64
```

15. The following command configure the NAT64 prefix:

```
ACOS(config)# cgnv6 nat64 prefix 64:ff9b::/64
```

16. The following command configure Fixed NAT inside users using IPv4/IPv6 inside user address with a session quota of 100:

```
ACOS(config)# cgnv6 fixed-nat inside 25.25.25.25 25.25.25.25 netmask
/32 nat 30.30.30.80 30.30.30.80 netmask /24 session-quota 100
ACOS(config)# cgnv6 fixed-nat inside 2001:db8::2:11 2001:db8::2:11
netmask 128 nat 30.30.30.90 30.30.30.90 netmask /24 session-quota 100
```

17. The following commands configure export of NetFlow records

```
ACOS(config)# netflow monitor n
ACOS(config-netflow-monitor)# record sesn-event-nat44 both
```

```
ACOS(config-netflow-monitor)# record sesn-event-nat64 both
ACOS(config-netflow-monitor)# destination 30.30.30.30
```

Client IP Address in Client HTTP Requests

This chapter provides information on how to configure the ACOS device to insert a client's IP address into the header of the client's HTTP request before the request is forwarded to the server. This configuration is useful when the source IP address of client requests is changed by NAT.

The following topics are covered:

Overview

The following information can help you configure the insertion of a client IP address in client HTTP requests:

- The ACOS configuration can contain a maximum of 32 HTTP-ALG templates.
- For NAT44 sessions, the inside (client) IPv4 address is inserted.
- For NAT64, 6rd-NAT64, or DS-Lite sessions the inside (client) IPv6 address is inserted.
- If you disable an HTTP-ALG template that is currently being used by active data sessions, it takes about 60 seconds for the template to be disassociated from the sessions.
- The supported HTTP methods are GET, HEAD, PUT, POST, OPTIONS, DELETE, TRACE, and CONNECT.
- This feature applies to NAT44, NAT64, 6rd-NAT64, DS-Lite, and Fixed-NAT. It and does not support 6rd or Static NAT sessions.
- This feature is not available for hairpin sessions.
- Querying external RADIUS servers is not support for Fixed-NAT.

Configuring IP Address Insertion in Client HTTP Requests

Configuring the Insertion of Client IP Addresses by Using the GUI

To configure HTTP-ALG Templates by using the GUI:

1. Navigate to **CGN > LSN > Template > HTTP-ALG**.
2. Click **Create**.
3. Enter a name.
4. In **Service Group**, select a service group.
5. Select the **Request Insert Client IP** check box.

When you select the check box, the client IP address is inserted in the client's HTTP request header.

6. In **Header Name Client IP** field, enter the new header name.

   By default, the client's IP address is inserted into the X-Forwarded-For header.

7. In **Method**, select one of the following options:

   - Replace
   - Append

8. Select the **Request Insert MSISDN** check box to specify whether to modify the MSISDN header of a client request by inserting the client's mobile number.
9. Click **Create**.

Configuring the Insertion of Client IP Addresses by Using CLI

Configuring the HTTP-ALG Template

1. Enter the following command to configure an HTTP-ALG template:

```
ACOS(config)# cgnv6 template http-alg ClientIP-Insert
```

2. Enter the following command to enable the insertion of the client IP address into the headers of the client's HTTP requests:

```
ACOS(config-http-alg:ClientIP-Insert)# request-insert-client-ip
ACOS(config-http-alg)# exit
```

Configuring the LSN Rule-list

In the rule-list, specify the destination IP addresses for which to perform client address insertion. For the action, specify **`template http-alg`** and the name of the HTTP-ALG template

1. Enter the following command to configure the LSN rule-list:

```
ACOS(config)# cgnv6lsn-rule-list RuleList1
```

2. Enter the following command to enter the configuration level for the default set of rules:

```
ACOS(config-lsn-rule-list)# default
ACOS(config-lsn-rule-list-default)# tcp port 80 action template http-alg ClientIP-Insert
ACOS(config-lsn-rule-list-default)# exit
ACOS(config-lsn-rule-list)# exit
```

NOTE: If the port range of the lsn-rule-list action template overlaps FTP ALG port 21 (for example), then the lsn-rule-list action template will have higher priority and FTP ALG won't be performed for port 21.

Adding the LSN Rule-list to the LSN LID

In the LSN LID, apply the LSN rule-list with the destination option.

1. Enter the following command to create or access the configuration level for the list:

```
ACOS(config)# cgnv6 lsn-lid 1
```

2. Enter the following command to bind the rule-list to the LID:

```
ACOS(config-lsn lid)# lsn-rule-list destination RuleList1
ACOS(config-lsn lid)# end
```

Displaying and Clearing ALG Statistics for HTTP

1. Enter the following command to display ALG statistics for HTTP:

```
ACOS# show cgnv6 http statistics
NAT HTTP-ALG Statistics:
----------------------------------------
HTTP Request Processed                  6
HTTP MSISDN Insertion Performed         0
HTTP Client IP Insertion Performed      6
Inserted MSISDN is 0000 (MSISDN Unavailable) 0
Queued Session Exceed Drop              0
MSISDN Query Succeed                    0
MSISDN Query Failed                     0
Query Request Sent                      0
Query Request Dropped                   0
```

```
Query Response Received                    0
Query Response Dropped                     0
```

The following entries are updated in the CLIENT IP address insertion output:

```
HTTP Request Processed 6
HTTP Client IP Insertion Performed 6
```

2. Enter the following command to clear ALG statistics for HTTP:

```
ACOS# clear cgnv6 http statistics
```

Client IP Insertion in HTTPS Requests on CGN/IPv6

This chapter provides information about how to configure the ACOS device to insert a client's IP address into the header of the client's HTTPS request before the request is forwarded to the server.

The following topics are covered:

Overview

The following information can help you configure the insertion of a client IP address in client HTTPS requests:

- This feature supports client IP insertion into HTTPS client requests similar to XFF feature already available.
- This feature supports only NAT64 and CGN/NAT44. Fixed-NAT NAT44, Fixed-NAT NAT64 and all other CGN IPv6Migration technologies are not supported.
- The feature takes advantage of HTTPS proxy in ADC/SLB using CGN pool instead of the regular SLB NAT pool.
- The **`allow-slb-cfg enable`** command allows SLB objects to be configured on a CGN partition for the sole purpose of supporting IP address insertion in HTTPS requests. This CLI command is not supported for any other SLB features. HTTPS virtual port must be configured along with a wildcard IPv4/IPv6 virtual server.
- The same user quota will be applied to both HTTPS traffic and CGN traffic for the same inside user.
- This feature supports CGN logging using CGN pool instead of the regular SLB NAT pool. The logging type supported is Syslog only. For more information about CGN logging, see *Traffic Logging Guide for IPv6 Migration.*

Configuring Client IP Insertion in HTTPS Requests

1. The following command enables the configuration of SLB objects in CGN partition:

```
ACOS(config)# allow-slb-cfg enable
```

NOTE: The command only supports this functionality. Prior to configuring any SLB objects, this command must be used in order to allow SLB objects to be configured on a CGN partition for the sole purpose of supporting IP address insertion in HTTPS requests.

2. The following commands configure class-lists for clients:

```
ACOS(config)# class-list nat64-clients
ACOS(config-class list)# 30::/64 lsn-lid 1

ACOS(config)# class-list lsn-clients
ACOS(config-class list)# 30.30.0.0/16 lsn-lid 1
```

3. The following commands configure an IPv4 access control list:

```
ACOS(config)# ip access-list acl4
ACOS(config-access-list:acl4)# permit ip any host 40.40.40.1
ACOS(config-access-list:acl4)# permit ip any host 40.40.40.2
```

4. The following commands configure an IPv6 access control list:

```
ACOS(config)# ipv6 access-list acl6
ACOS(config-access-list:acl6)# ipv6 access-list acl6
ACOS(config-access-list:acl6)# permit ipv6 any 64:40::/96
```

5. The following commands create the Ethernet interfaces connected to the firewalls and the real servers or clients, and then enable promiscuous mode:

```
ACOS(config)# interface ethernet 1
ACOS(config-if:ethernet:1)# enable
ACOS(config-if:ethernet:1)# ip address 30.30.30.80 255.255.255.0
ACOS(config-if:ethernet:1)# ip allow-promiscuous-vip
ACOS(config-if:ethernet:1)# ip nat inside
ACOS(config-if:ethernet:1)# ipv6 address 30::88/64
ACOS(config-if:ethernet:1)# ipv6 nat inside

ACOS(config)# interface ethernet 2
ACOS(config-if:ethernet:2)# enable
ACOS(config-if:ethernet:2)# ip address 40.40.40.80 255.255.255.0
ACOS(config-if:ethernet:2)# ip nat outside
```

6. The following commands configure a server SSL template to use the certificate and key:

```
ACOS(config)# slb template server-ssl sssl1
ACOS(config-server ssl)# cert server.crt
ACOS(config-server ssl)# key server.key
```

7. The following commands configure a real server:

```
ACOS(config)# slb server rs1 40.40.40.121
```

```
ACOS(config-real server)# port 443 tcp
ACOS(config-real server-node port)# exit
ACOS(config-real server)# exit
```

```
ACOS(config)# slb server rs2 64:40::2828:2879
ACOS(config-real server)# port 443 tcp
ACOS(config-real server-node port)# exit
ACOS(config-real server)# exit
```

8. The following commands configures the service group:

```
ACOS(config)# slb service-group sg3 tcp
ACOS(config-slb svc group)# health-check-disable
ACOS(config-slb svc group)# member rs1 443
ACOS(config-slb svc group-member:443)# exit
ACOS(config-slb svc group)# exit

ACOS(config)# slb service-group sg4 tcp
ACOS(config-slb svc group)# member rs2 443
ACOS(config-slb svc group-member:443)# exit
ACOS(config-slb svc group)# exit
```

9. The following commands configure the client SSL template:

```
ACOS(config)# slb template client-ssl cssl1
ACOS(config-client ssl)# cert server.crt
ACOS(config-client ssl)# key server.key
```

10. The following commands configure the HTTP template:

```
ACOS(config)# slb template http http1
ACOS(config-http)# insert-client-ip X-Forwarded-For replace
```

11. The following commands configure the virtual server:

```
ACOS(config)# slb virtual-server vs3 0.0.0.0 acl name acl4
ACOS(config-slb vserver)# port 443 https
ACOS(config-slb vserver-vport)# no-dest-nat
ACOS(config-slb vserver-vport)# source-nat use-cgnv6
ACOS(config-slb vserver-vport)# service-group sg3
ACOS(config-slb vserver-vport)# template http http1
ACOS(config-slb vserver-vport)# template server-ssl sssl1
```

```
ACOS(config-slb vserver-vport)# template client-ssl cssl1
```

```
ACOS(config)# slb virtual-server vs4 :: ipv6-acl acl6
ACOS(config-slb vserver)# port 443 https
ACOS(config-slb vserver-vport)# no-dest-nat
ACOS(config-slb vserver-vport)# source-nat use-cgnv6
ACOS(config-slb vserver-vport)# service-group sg3
ACOS(config-slb vserver-vport)# template http http1
ACOS(config-slb vserver-vport)# template server-ssl sssl1
ACOS(config-slb vserver-vport)# template client-ssl cssl1
```

12. The following command binds the class list to the LSN feature:

```
ACOS(config)# cgnv6 lsn inside source class-list lsn-clients
```

13. The following command configures CGN pools:

```
ACOS(config)# cgnv6 nat pool cgnp1 40.40.40.100 40.40.40.100 netmask
/24
```

14. The following commands configure a LSN_LID and add the pool to it:

```
ACOS(config)# cgnv6 lsn-lid 1
ACOS(config-lsn-lid)# source-nat-pool cgnp1
ACOS(config)# exit
```

15. The following command binds the class list to the NAT64 feature:

```
ACOS(config)# cgnv6 nat64 inside source class-list nat64-clients
```

16. The following command configures the NAT64 prefix:

```
ACOS(config)# cgnv6 nat64 prefix 64:40::/96
```

Client Mobile Numbers in Client HTTP Requests

This chapter explains how to configure the ACOS device to obtain a client's mobile number from a RADIUS server, and how to insert this mobile number in the X-MSISDN header of a client's request and forward the request to the server.

The following topics are covered:

Overview

This feature is useful for tracking clients who are paid subscribers to a mobile HTTP service. When a mobile client sends an HTTP request, the billing server for the paid service can retrieve the mobile number from the header of the client's HTTP request. This enables the billing application to correctly distinguish paid subscribers.

Most applications check for a client's mobile number in the X-MSISDN header of the client request. The following section describes how to configure the ACOS device to obtain client mobile numbers from RADIUS servers insert the numbers in the X-MSISDN headers in a client request.

Consider the following information:

- This feature supports NAT44, NAT64, 6rd-NAT64, and DS-Lite and does not support 6rd, Static NAT, or Fixed-NAT sessions.
- This feature is not available for hairpin sessions.
- For NAT44 sessions, the inside (client) IP address used for a query is always an IPv4 address. For NAT64, 6rd-NAT64, and DS-Lite sessions, the inside (client) IP address used for a query is always an IPv6 address.
- In the current release, the ACOS device can run a maximum of 40,000 simultaneous query sessions. Sessions which exceed this value are automatically dropped.
- This feature supports the following HTTP methods: GET, HEAD, PUT, POST, OPTIONS, DELETE, TRACE, and CONNECT.
- This feature is independent of the feature for inserting client MSISDN values into CGN traffic logs. (See the *Traffic Logging Guide for IPv6 Migration*).

Remember the following issues:

- The service group containing the client RADIUS servers must use the round-robin load-balancing method.
- The ACOS configuration can contain a maximum of 32 HTTP-ALG templates.
- If you disable an HTTP-ALG template that is currently in use by active data sessions, it takes around 60 seconds for the template to be disassociated from the sessions.

- The maximum length supported for the mobile phone number value is 15 digits.
- You can not manually clear RADIUS query sessions.
- If the query fails, a fake MSISDN with the value “0000” is automatically inserted in the HTTP header.
- Each inside IP address (client) must query at least once when the IP address has a session through the ACOS device. If all client session timeout, another query automatically occurs.

ACOS RADIUS Server

To obtain a client’s mobile number, the ACOS device acts as a RADIUS client. The ACOS device sends a ADIUS accounting-request message to the RADIUS server used by the clients requesting the MSISDN from the RADIUS server. The ACOS device inserts the mobile number that is received in the accounting-reply message in the X-MSISDN header of the client’s request before forwarding the request to the content server.

Example of RADIUS Accounting-request Messages Sent by ACOS Device

Here is an example of a RADIUS accounting-request message sent by the ACOS device to a client’s RADIUS server for a NAT44 client.

```
Accounting Request {
      Header : {
             Packet Code=ACCT_REQ (1 octet)
             Id=XXX (1 octet)
             Length = XXX (2 octets)
             Request Authenticator = 0123456789ABCDEF (16 octets)
      }
      Attributes : {
             Acct-Status-Type : Integer Value = 1 (4 octets)
             Acct-Session-Id : String Value = DF  (>=1 octet)
             NAS-ID : String Value = ACOS@1.2.3.4 (MAX 50 octets)
             Vendor-Specific: String Value= {
                 A10-CGN-Action: Integer Value = A10-CGN-QUERY (4 octets)
                 A10-CGN-Inside-Addr : String Value = 192.168.150.100 (4
octets)
             }
```

```
        }
}
```

The Request Authenticator is a secure hash value that the RADIUS server and ACOS device use to authenticate the RADIUS traffic that flows between them.

The following vendor-specific attributes are included:

- A10-CGN-Action – The type of RADIUS message (A10-CGN-QUERY in this case)
- A10-CGN-Inside-Addr – The inside client's IP address
- For DS-Lite, NAT64, or 6rd-NAT64 clients, the inside client's IPv6 address is used for a query.

For more information, see the "Management Security Features" chapter in the System Configuration and Administration Guide.

A Successful Reply Message from Client RADIUS Server

The sections below provide examples of a successful (and an unsuccessful) reply to a RADIUS accounting-request message from the ACOS device.

NOTE: Remember the following issues:
- If the RADIUS server finds the MSISDN for the inside IP address of the ACOS device, and the response does not carry an A10-CGN-Response code and Calling-Station-ID, the ACOS device will drop the response.
- The ACOS device accepts the MSISDN only as a numeric string (for example, "6086227037"). Values carried in the Calling-Station-ID that do not follow this format are automatically dropped.

Accounting Response Message

```
Accounting Response {
      Header : {
             Packet Code=ACCT_RESP (1 octet)
             Id=XXX (1 octet)
             Length = XXX (2 octets)
             Request Authenticator = 0123456789ABCDEF (16 octets)
      }
      Attributes : {
             Vendor-Specific: String Value= {
```

```
                    A10-CGN-Response: Integer Value = SUCCESS (4 octets)
                }
                Calling-Station-Id =16086171111 (MAX 15 octets)
        }
}
```

The **Calling-Station-Id** field contains the client's mobile number.

NOTE: This is the Type 31 attribute in RFC 2865.

An Unsuccessful Reply Message from Client RADIUS Server

Here is an example of an unsuccessful reply to a RADIUS accounting-request message from the ACOS device.

NOTE: If the RADIUS server is unable to find the MSISDN corresponding to the inside IP address from the ACOS device, the response must carry an A10-CGN-Response code.

```
Access Response {
        Header : {
                Packet Code=ACCT_RESP (1 octet)
                Id=XXX (1 octet)
                Length = XXX (2 octets)
                Request Authenticator = 0123456789ABCDEF (16 octets)
        }
        Attributes : {
                Vendor-Specific: String Value= {
                    A10-CGN-Response: Integer Value = FAILURE (4 octets)
                }
        }
}
```

In this example, the value of the A10-CGN-Response field is FAILURE. (See the "Management Security Features" chapter of the System Configuration and Administration Guide.)

NOTE: A successful A10-CGN-Response returns an integer value of 1, and a failed A10-CGN-Response returns an integer value of 2.

Configuring Insertion of Client Mobile Numbers in Headers of HTTP Requests

You can configure the insertion of client mobile numbers in the HTTP request headers by using the GUI and CLI.

Configuring the Insertion of Client Mobile Numbers Using the GUI

Adding the IP List for the Client RADIUS Servers

NOTE: Use this procedure or the one in Adding the Server Configurations and Service Group for the Client RADIUS Servers to define the RADIUS servers.

To add an IP list for client RADIUS servers:

1. Navigate to **CGN > LSN > Class Lists**.
2. Click **Create**.
3. Enter a Name for the class list.
4. Select **IPv4** or **IPv6**.
5. To configure an IPv4/IPv6 list, enter the parameters for the corresponding fields.
6. Click **Create**.

Adding the Server Configurations and Service Group for the Client RADIUS Servers

To create a server configuration for a client RADIUS server:

1. Navigate to **CGN > Services > Service Groups**.
2. Click **Create**.
3. Enter a name for the group of servers.
4. In **Protocol**, select **UDP**.

5. In the **Member** section, click **Create**.
6. Select a configured server from the Server drop-down list or click New Server to display configuration fields for the server.
7. Select the address type and enter the hostname or IP address.
8. In the **Port** field, enter the server's UDP port.
9. Click **Create Member**.
10. Click **Update Service Group**.

Configuring the HTTP-ALG Template

To configure an HTTP-ALG template for mobile number insertion:

1. Navigate to **CGN > LSN > Template > HTTP-ALG**.
2. Click **Create**.
3. Enter name for the template.
4. Select the **Request Insert MSISDN** check box.

   When you select the check box, the client mobile number is inserted in the client's HTTP request header.

5. In **Header Name MSISDN** field, enter the new header name.

   By default, the client's IP address is inserted into the X-MSISDN header.

6. Select the RADIUS service group from the **RADIUS service group** drop-down list.
7. In **RADIUS Server Maximum Retries**, enter the number of times that the ACOS device can resend a request that times out.

   The default is 2, and you can enter between 0-3.

8. In **Maximum RADIUS Server**, select whether the ACOS device is allowed to send a timed out query to a different server in the service group.

   You can enter 0 (disabled) or 1 (try up to one additional server). If you enter 1, but the service group contains only one server, the ACOS device creates another session with the same server.

9. In **Timeout**, enter the maximum number of seconds the ACOS device waits for a reply to a RADIUS accounting-request message from the ACOS device to the client RADIUS server. You can specify 1-3 seconds. The default is 2.
10. Click **Create**.

Configuring the Insertion of Client Mobile Numbers Using the CLI

Add RADIUS Server Information

Method 1: Configure an IP list that contains the IP address of each RADIUS server

NOTE: Use this procedure or the one in Adding the Server Configurations and Service Group for the Client RADIUS Servers to define the RADIUS servers.

Using this method, ACOS acts as a RADIUS server, and receives RADIUS Accounting information from external RADIUS servers. ACOS caches the numbers and can insert these numbers in CGN log messages, HTTP requests (or both) based on your configuration.

This method does not require configuration of a service group for the external RADIUS servers. Service-group configuration for querying external RADIUS servers is optional. Queries are performed only when ACOS does not have the MSISDN of the client in the cache.

NOTE: The MSISDN obtained from queries is not stored in the cache.

Commands

1. Enter the following command to create an IP list for client RADIUS servers:

```
ACOS(config)# ip-list RADIUS_IP_LIST
ACOS(config-ip list)# 9.9.9.9 to 9.9.9.10
ACOS(config-ip list)# exit
```

2. The following commands configure RADIUS server parameters for ACOS:

```
ACOS(config)# system radius server
```

```
ACOS(config-lsn radius)# remote ip-list RADIUS_IP_LIST
ACOS(config-lsn radius)# secret a10rad
ACOS(config-lsn radius)# listen-port 1813
ACOS(config-lsn radius)# attribute inside-ip number 8
ACOS(config-lsn radius)# attribute msisdn number 31
ACOS(config-lsn radius)# exit
```

Method 2: Add a server configuration for each client RADIUS server, and add them to a service group

The following commands configure insertion of client mobile numbers into the headers of client requests:

To begin, the following commands create configurations for the client RADIUS servers and add them to a service group:

1. Enter the following command to create a server configuration for a client RADIUS server:

```
ACOS(config)# cgnv6 server radius1 203.0.118.2
```

2. Enter the following command to sspecify the UDP port on which the server listens for RADIUS accounting traffic:

```
ACOS(config-real server)# port 1813 udp
ACOS(config-real server-node-port)# exit

ACOS(config)# cgnv6 server radius2 192.168.1.2
ACOS(config-real server)# port 1813 udp
ACOS(config-real server-node-port)# exit
```

3. Enter the following command to create a service-group (server pool) for the traffic log servers:

```
ACOS(config)# cgnv6 service-group RADIUS_SVG udp
```

NOTE: A service group is required even if you plan to use only a single client RADIUS server.

4. Enter the following command to add a client RADIUS server and its UDP port to the service group:

```
ACOS(config-cgnv6 svc group)# member radius1 1812
ACOS(config-cgnv6 svc group-member:1812)# member radius2 1812
ACOS(config-cgnv6 svc group-member:1812)# exit
```

The following commands configure the HTTP-ALG template for insertion of client mobile numbers in HTTP requests:

Configure an HTTP-ALG Template

When configuring an HTTP-ALG template, you can specify the following options:

- Enable client mobile number insertion.
- Specify the name of the IP list or service group that contains the client RADIUS servers to the template.
- Specify the shared-secret password string to use for authenticating RADIUS traffic between the ACOS device and the client RADIUS servers.

To configure an HTTP-ALG template for mobile number insertion, use the following commands:

1. Enter the following command to create the template:

```
ACOS(config)# cgnv6 template http-alg CLIENT-MOBILE-TEMPLATE
```

2. Enter the following command to insert Client IP into HTTP request

```
ACOS(config-http-alg:template1)# request-insert-msisdn header-name
header1
```

NOTE:

- All retry attempts are used with the first server before the next RADIUS server is tried. The number of retry attempts is defined as the sum of values for the **retry** and **retry-svr-num** configuration options.
- For example, if the value **retry** is set to 3 and **retry-svr-num** is set to 1, the ACOS device will try the first server 4 times (1+3 retry attempts), and try a second server 4 times (1+3 retry attempts).
- If all servers are down, the query will fail after completing the entire retry process.

Configure an LSN Rule-list

The rule-list, specify the client IP addresses for which to perform mobile number insertion. For the action, specify “template http-alg” and the name of the HTTP-ALG template.

The following commands configure the LSN rule-list:

1. Enter the following command to configure header insertion in an LSN rule-list:

```
ACOS(config)# cgnv6 lsn-rule-list rule1
```

2. Enter the following command to enter the configuration level for the set of rules to apply to the specified IP host address or subnet:

```
ACOS(config-lsn-rule-list)# ip 10.3.3.0/24
ACOS(config-lsn-rule-list-ip)# tcp port 80 action template http-alg
CLIENT-MOBILE-INSERT
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# ip 10.1.1.1/32
ACOS(config-lsn-rule-list-ip)# tcp port 8080 action template http-alg
CLIENT-MOBILE-INSERT
ACOS(config-lsn-rule-list-ip)# exit
ACOS(config-lsn-rule-list)# exit
```

Adding the LSN Rule-list to the LSN LID

Enter the following commands to add the LSN rule-list to the LSN LID:

1. Enter the following command to create or access the configuration level for the list:

```
ACOS(config)# cgnv6 lsn-lid 4
```

2. Enter the following command to bind the rule-list to the LID:

```
ACOS(config)# cgnv6 lsn-rule-list rule1
ACOS(config-lsn-rule-list)# exit
ACOS(config)# cgnv6 lsn-lid 4
ACOS(config-lsn-lid)# lsn-rule-list destination rule1
```

Displaying and Clearing ALG Statistics for HTTP

1. The following command shows statistics for HTTP ALG:

```
ACOS# show cgnv6 http-alg statistics
NAT HTTP-ALG Statistics:
-----------------------------------------------
HTTP Request Processed                          0
HTTP MSISDN Insertion Performed                 0
HTTP Client IP Insertion Performed              0
Inserted MSISDN is 0000 (MSISDN Unavailable)    0
Queued Session Exceed Drop                      0
MSISDN Query Succeed                            0
MSISDN Query Failed                             0
Query Request Sent                              0
Query Request Dropped                           0
Query Response Received                         0
Query Response Dropped                          0
```

2. Enter the following command to clear ALG statistics for HTTP:

```
ACOS# clear cgnv6 http-alg statistics
```

Fixed-NAT

This chapter provides an overview of Fixed-NAT and how to configure it.

The following topics are covered:

Overview

Fixed-NAT allocates NAT ports for each client from a predetermined set of ports on the NAT address. Since each client that is using Fixed-NAT gets a fixed set of ports, a client can be identified without a log. A client can be identified based solely on the NAT IP address and the port numbers in the client's fixed allocation of ports.

NOTE: A NAT64 prefix with mapping to a class list is not supported for Fixed NAT.

You can also configure a dynamic pool of ports to provide additional ports to clients who run out of NAT ports. Since the dynamic pool of ports can be used by any client, logging still is applicable to port allocations from the dynamic pool.

NOTE: The EIM/EIF is the same for all NAT44, NAT64, DS-Lite, and Fixed-NAT. The only difference is in the allocation of the NAT IP addresses and ports. For more information, see Full-Cone NAT.

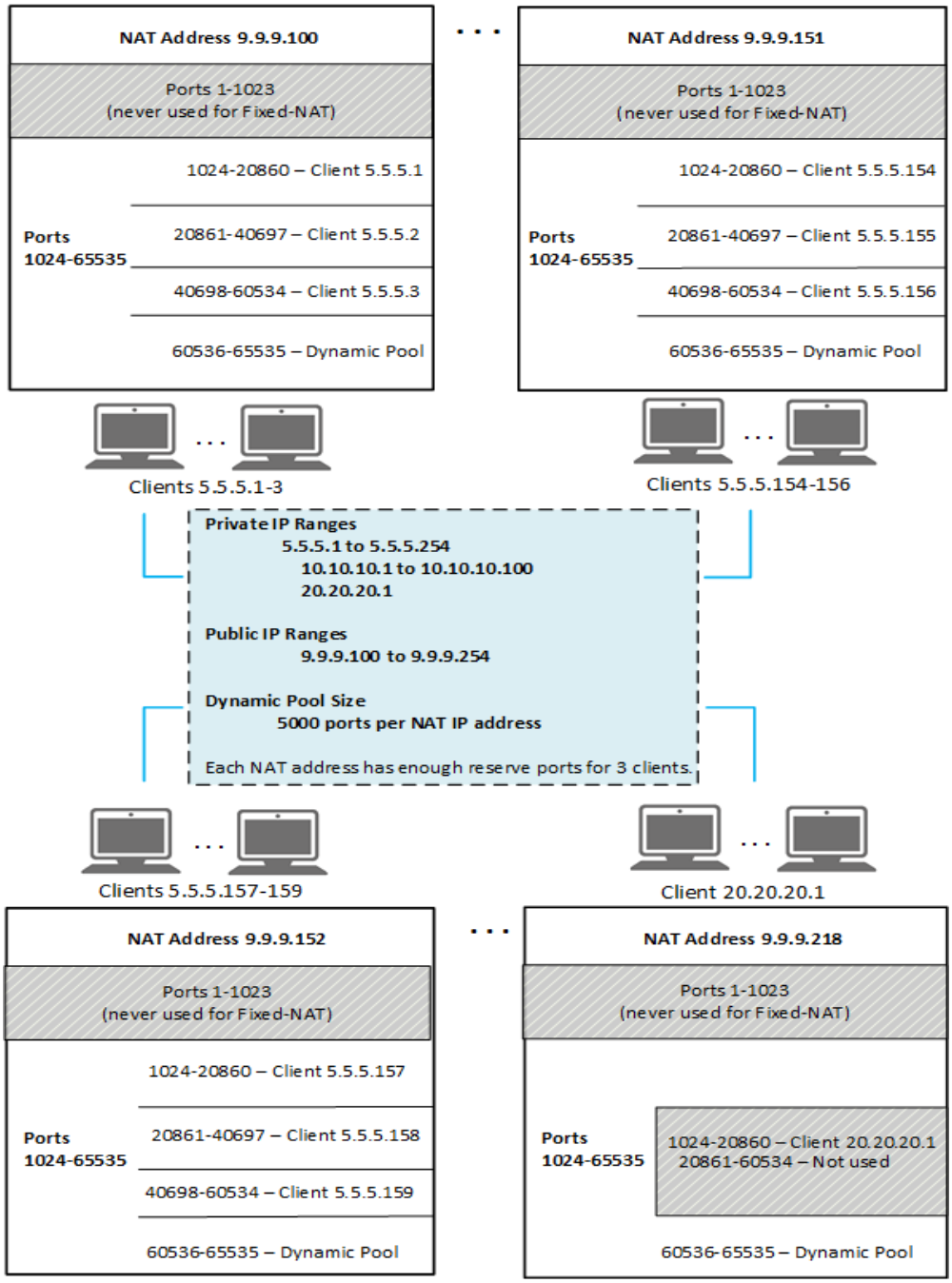
ALG support for Fixed-NAT applies to the following protocols:

- ESP
- FTP
- TFTP
- RTSP
- PPTP
- SIP

Fixed-NAT

Figure 27 illustrates an example of a Fixed-NAT deployment for multiple client IP ranges.

Figure 27 : Fixed-NAT with multiple client ranges



In this example, Fixed-NAT is configured for the following client IP ranges:

- 5.5.5.1 to 5.5.5.254 – 254 client addresses
- 10.10.10.1 to 10.10.10.100 – 100 client addresses
- 20.20.20.1 – a single client address

All the clients are mapped to NAT addresses in the 9.9.9.100-254 range.

NOTE: For simplicity, the figure shows the mappings for only some client addresses.

Port assignments are calculated by using the number of inside clients and the number of NAT addresses and the following operations:

- The ACOS device divides the number of clients by the number of NAT addresses.
- The ACOS device divides the number of available ports per NAT address by the number of clients per NAT address.

For example, in Figure 27, there are 355 inside clients and 155 NAT addresses. Rounding up, this is 355 / 155 = 3 inside clients per NAT address. With 3 inside clients per NAT address, only 119 NAT addresses are needed and the rest remain unused.

On each NAT address, by default, 64512 protocol ports are available for client mappings. (Ports 1-1023 are never used for Fixed-NAT.) You can also configure a lower range of ports as usable NAT ports.

In this example, 5000 ports are set aside on each NAT address as a dynamic pool of ports. The pool of ports is used by inside clients who run out of reserved ports. This leaves 59512 ports that can be reserved for individual client addresses. For more information, see Dynamic Pools.

Three inside clients per NAT address and 59512 ports per NAT address results in 19837 ports per inside client.

NOTE: This example assumes all the ports can be used for Fixed-NAT. You can also explicitly specify the range of usable NAT ports.

If a NAT address has leftover ports, but not enough ports for another client, the leftover ports are unused. In Figure 27, each NAT address has enough ports to provide 19837 port ranges to 3 clients. As a result, 59511 ports are used with 1 port left over (60535).

Protocol Port Use

Only “ephemeral” ports (1024-65535), and not well-known ports (1-1023), can be used for Fixed-NAT mappings.

By default, Fixed-NAT uses ports in the 1024-65535 range. You can change this port range when you create the simplified Fixed-NAT configuration.

Dynamic Pools

A dynamic pool is a range of IP addresses that is set aside for clients who do not have any available reserved ports.

For example, if client 5.5.5.1 is already using ports 1024-26839, and the client needs more ports, the additional ports can be allocated from the dynamic pool (in this example, 60536-65535).

NOTE: The dynamic pool ports on a NAT address are available only to clients that are mapped to that address. For example, dynamic pool ports on NAT address 9.9.9.100 can be used by clients 5.5.5.1 and 5.5.5.2 but not by any other clients.

Port Allocation Logic

Allocating Fixed-NAT ports depends on the Fixed-NAT configuration.

Port allocation is completed in the following way:

1. Calculate the ports-per-user.
2. Get the usable NAT port range, which, by default, is 1024-65535.

   This range can be different if a dynamic pool is configured, or the range of usable NAT ports is set.
3. Sequentially allocate port ranges to inside clients, in order of ascending client IP address.

   The port range sizes are based on the ports-per-user settings.

Available NAT ports in each NAT address

Consider the following information about NAT ports:

- If you do not configure the number of usable NAT ports, ports 1024-65535 (U) are available on each NAT address.
- If you do not set the configured dynamic pool size (D) to 0, the number of available NAT ports is (N) = U - D.

Calculating Ports-per-user

To calculate ports-per-user:

1. Calculate total number of inside client users (I).
2. Calculate total number of Public NAT addresses (P).
3. Calculate number of Inside clients per NAT address (T)= roundup(I/P).
4. Calculate ports-per-user = N / T.

Fixed-NAT Address Mapping Methods

You can select one of the following options to map inside client IP addresses to public NAT IP addresses:

- Use Least NAT IPs—Inside client IP addresses can be allocated to NAT addresses with the goal of minimizing the use of available public NAT IP Addresses. This is the current behavior. This configuration method may result in some unused NAT IP addresses. For details, refer to Use Least NAT IPs.
- Use All NAT IPs—Inside client IP addresses can be allocated with the intent to use all of the available NAT IP addresses. This new algorithm ensures that all NAT IP addresses are used, with little room for any unused NAT IP addresses. For details, refer to Use All NAT IPs with an Offset.

By default, if neither method is explicitly configured, use the Use Least NAT IPs method.

You can also configure an “offset” when you map an inside client IP address to an external NAT IP address. By default, the first inside client IP address is automatically mapped to the first NAT IP address. However, with the option to specify the offset, the ACOS software allows you to indicate the first inside client IP addresses to any NAT IP address to which the offset points.

Use Least NAT IPs

Inside client IP addresses are mapped sequentially to the available NAT IP addresses. When a NAT IP address has reached the maximum number of inside client IP addresses that it can support, the remaining inside client IP addresses are mapped to the next NAT IP address. The goal is to use the least number of public NAT IP addresses, so this process ensures that each NAT IP address is completely used before the next NAT IP address is used.

If you do not select a Fixed-NAT address mapping method, by default, the **`use-least-ip`** option is used.

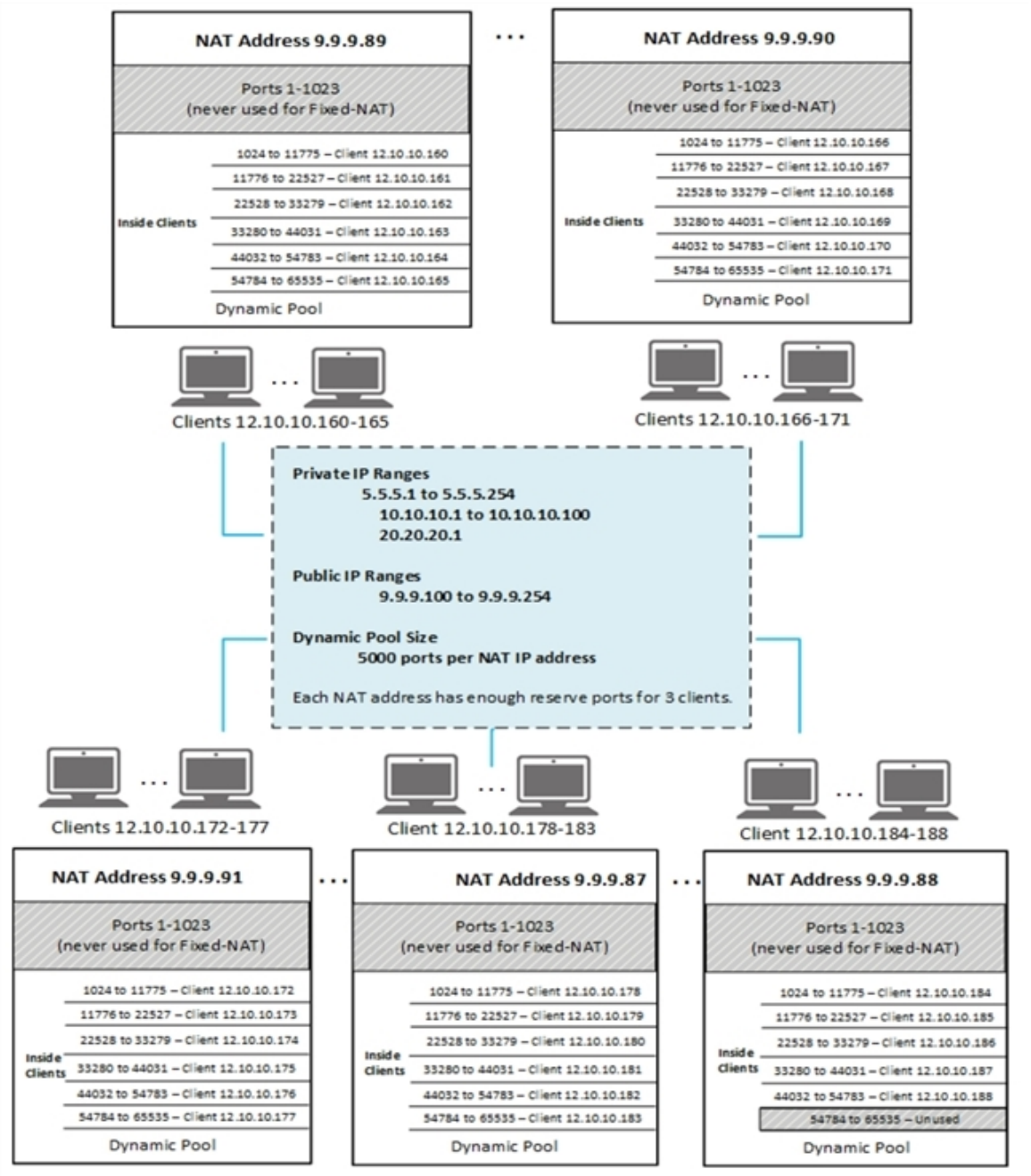
Use Least NAT IPs with an Offset

This figure illustrates the process of using the least number of NAT IPs. As shown, each NAT IP address supports six inside client IP addresses. In this example, the six inside client IP addresses are mapped to five different NAT IP addresses, leaving part of the fifth NAT IP address unused. However, note that the offset of two causes the first inside client (12.10.10.160) to be mapped to NAT IP address (9.9.9.89). Sequentially, six client IPs are mapped per NAT IP address, except for the last NAT IP that only contains five client IP entries.

Use All NAT IPs with an Offset

You can also use the all NAT IP address mapping model. In contrast to the use least NAT IP address mapping model, each of the first five inside client IP addresses are mapped to each NAT IP address. After the first five addresses are mapped, the sixth address is mapped again to the same NAT IP address as the first inside client address and so on. The offset of **2** causes the first inside client (12.10.10.160) to be mapped to NAT IP address (9.9.9.89) and the second (12.10.10.161) to the second NAT IP address (9.9.9.90). The cycle continues until all client IP addresses are allocated, which leaves no unused NAT IP addresses.

Figure 28 : Using the Least NAT IPs with an Offset (Value of 2)



Fixed-NAT Configuration Options

You can configure the following Fixed-NAT options:

- Inside IP list – Name of IP list that contains the ranges for IPv4/IPv6 inside clients. This also can be one IPv4/IPv6 range. You can specify one contiguous address range or the name of an IP list. To configure Fixed-NAT for multiple address ranges, you must use an IP list.

NOTE:

- The maximum number of IP lists or IP ranges varies depending on your Thunder Series model.
- The maximum number of IP list entries or IP ranges that can be added to the fixed-NAT configuration in a partition is 10,000 for platforms below 64GB, 20,000 for a 64GB platform, and 30,000 for a 128GB platform.
- Also, the maximum number of IPv4 or IPv6 addresses, ranges, or prefixes supported per IP list is 1024 for platforms below 64GB, 2048 for a 64GB platform, and 3072 for a 128GB platform.

- Outside IP list – Name of IP list that contains the ranges for NAT addresses. This also can be one NAT address range.
- (Optional) Usable NAT ports – Range of protocol ports that can be allocated to clients. You can specify a protocol port in the 1024-65535 range.
- (Optional) Ports per user – Number of protocol ports to allocate to each new client. You can specify a port in the 1-64512 range. If you do not specify this option, the ACOS device automatically calculates it based on the configuration. For more information, see Port Allocation Logic.
- (Optional) Dynamic pool size – Number of addresses to set aside for clients that run out of reserved ports. The dynamic pool is allocated from the top of the range of usable NAT ports.

  For example, if the range of usable NAT ports is 5000-60000, and the dynamic pool size is 5000, ports 55001-60000 are allocated to the dynamic pool. Only ports 5000-55000 can be reserved for clients. By default, there is no dynamic pool.
- (Optional) Session quota – Maximum number of sessions that can be created for a client. You can specify 1-2147483647. By default, there is no session quota.
- (Optional) VRRP-A VRID – ID of the VRRP-A virtual router to which the Fixed-NAT addresses must be assigned. There is no default.

IPv4 Inside Clients Configuration Workflow

Here is a high-level view of the process to configure Fixed-NAT for multiple client IPv4 address ranges:

1. Configure an IP list that specifies the inside (private) IPv4 ranges.
2. Configure an IP list that specifies the outside (public) IP ranges.
3. Configure Fixed-NAT to use the IP lists together to create the fixed IP address mappings. Optionally, you also can configure the following Fixed-NAT options for the IP ranges:

- Usable NAT ports
- Ports per user
- Dynamic pool size
- Session quota
- HA group ID

IPv6 Inside Clients Configuration Workflow

Here is a high-level view of the process to configure Fixed-NAT for multiple client IPv6 address ranges, such as for DS-Lite:

1. Configure an IP list that specifies the inside (private) IPv6 ranges.
2. Configure an IP list that specifies the outside (public) IP ranges.
3. Configure Fixed-NAT to use the IP lists together to create the fixed IP address mappings. Optionally, you also can configure the following Fixed-NAT options for the IP ranges:

- Usable NAT ports
- Ports per user
- Dynamic pool size
- Session quota
- VRRP-A VRID

For NAT64, you can map public IPv4 NAT addresses to the IPv6 prefixes or individual IPv6 addresses of the subscriber. For DS-Lite, you can map public IPv4 NAT addresses to the IPv6 prefixes or individual IPv6 addresses of Customer Premises Equipment (CPE). The CPE is the DS-Lite router in the DS-Lite customer's home network. The CPE encapsulates the client's IPv4 traffic into an IPv6 tunnel for transport between the customer's home network and the customer's ISP.

The address allocation for DS-Lite and NAT64 Fixed-NAT clients works the same as the Fixed-NAT address allocation for IPv4 clients, such as the Fixed-NAT address allocation for IPv4 clients, such as LSN clients. For more information, see Configuring Fixed NAT.

The same optional parameters, such as dynamic pool size, HA group, and session quota are supported. LSN user quotas also are supported.

NOTE:

- For NAT64 Fixed-NAT, user quotas are calculated based on the subscriber's IPv6 prefix or address.
- For DS-Lite Fixed-NAT, user quotas are calculated based only on the CPE IPv6 prefix or address. All IPv4 addresses using the same IPv6 CPE address are classed together as one inside client who is assigned one NAT IP address and one port range.

CPE Ranges

Consider the following information:

- To apply Fixed-NAT to ranges of IPv6 CPE prefixes, instead of individual CPE addresses, use an IP list to specify the CPE prefix range.
- To specify CPE prefixes, you must use an IP list.
- To specify individual CPE addresses, you can use an IP list or specify the addresses when you enable Fixed-NAT.

If you configure Fixed-NAT for IPv6 CPE prefixes, instead of individual addresses, all IPv6 clients that share the same prefix receive one NAT IP address and one port range.

NOTE: An IP list can contain up to 1024 IPv6 prefixes.

Application Level Gateway (ALG) Support

ALG is supported for DS-Lite Fixed-NAT traffic for the following protocols:

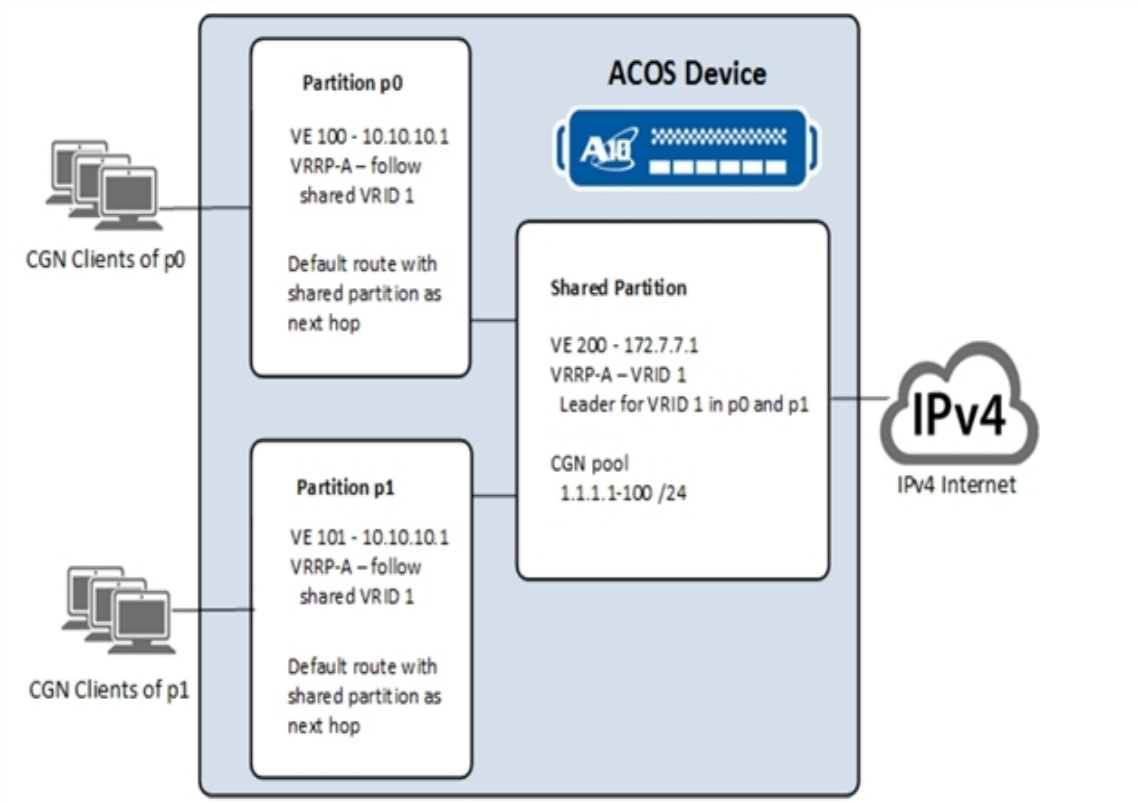
- FTP
- TFTP
- RTSP
- PPTP
- SIP

L3V Inter-partition Routing for Fixed-NAT

ACOS includes L3V support for NAT44, NAT64 and DS-Lite in the following scenarios:

- Exclusively in shared partition
- Exclusively in private partition(s)
- Inter-partition with the following:
  - Inside clients connected to a private partition
  - Shared partition connecting to the IPv4 Internet

Figure 29 : Fixed-NAT with Inter-Partition Routing



This example has two private partitions with L3V enabled. Each partition is connected to its own set of Fixed-NAT clients that share overlapping private IPv4 addresses. The shared partition is connected to the IPv4 internet.

The Fixed-NAT configurations for inter-partition routing need to be made in the shared partition. Client traffic is received by the private partitions on their VLAN Virtual Ethernet (VE) interfaces. The incoming traffic is handled based on the Fixed-NAT configuration in the shared partition. Each private partition has a default route, where the next hop is the shared partition.

VRRP-A is used for redundancy. (The second ACOS device is not shown.) The Fixed-NAT configurations are backed up by VRRP-A. Each private partition is configured to base its VRRP-A Active/Standby state on the state of the shared partition's VRID.

Configuring Fixed NAT

Configuring IPv4 Inside Clients

Configuring IPv4 Inside Clients by Using the GUI

You can configure IPv4 inside clients by using the GUI.

Configuring IP Lists

To configure IP lists:

1. Navigate to **CGN > Fixed NAT > IP Lists**.
2. Click **Create**.
3. Enter a Name for the list.
4. Select the **IP** type.
5. Click **Add**.
6. In **Start Address**, enter the beginning (lowest) IP address in the range.
7. In **End Address**, enter the ending (highest) IP address in the range.
8. Repeat these steps for each IP address range, if applicable.
9. Click **Create**.

Configuring Fixed NAT

To configure Fixed NAT:

1. Navigate t**o CGN** > **Fixed NAT**.
2. Click **Create**.
3. In the **Inside** section, complete the following steps:

NOTE: You can specify one, contiguous IP range or an IP list.

- To specify one IP range:
  a. Select **IP Address**.
  b. Select **IPv4** or **IPv6**.
  c. Enter the Start Address, End Address and Netmask of the range.

  To specify an IP list:

  i. Select **IP List**.
  ii. Select **IPv4** or **IPv6**.
  iii. In **IP list**, select the IP list.

4. In **NAT**, complete the following steps:

NOTE: You can specify one, contiguous IP range or an IP list.

- To specify an **IP range**:
  a. In **Inside**, select **IPv4** or **IPv6**.
  b. In **Start Address**, enter the beginning (lowest) IP address in the range.

  To specify an IP list:

  i. In the **Inside** section, select **IPv4** or **IPv6**.
  ii. In the **IP list** drop-down list, select the IP list.

5. Configure general settings, if applicable. Click **General Fields** to display the options. (For more information, see Fixed-NAT Configuration Options.)
6. Click **Create**.

Configuring IPv4 Inside Clients by Using the CLI

The following commands implement the Fixed-NAT deployment shown in Figure 27.

1. Enter the following command to configure the IP list for the client IP address ranges:

```
ACOS(config)# ip-list fixed-nat-inside-users
```

2. Enter the following commands to configures an IP range based on the beginning

and ending host address:

```
ACOS(config-ip list)# 5.5.5.1 to 5.5.5.254
ACOS(config-ip list)# 10.10.10.1 to 10.10.10.100
ACOS(config-ip list)# 20.20.20.1
ACOS(config-ip list)# exit
```

3. Enter the following commands to configure the IP list for the NAT IP address range:

```
ACOS(config-ip list)# ip-list fixed-nat-public-address
ACOS(config-ip list)# 9.9.9.100 to 9.9.9.254
ACOS(config-ip list)# exit
```

4. Enter the following command to configure Fixed-NAT for multiple client IPv4 address ranges:

```
ACOS(config)# cgnv6 fixed-nat inside ip-list fixed-nat-inside-users nat
ip-list fixed-nat-public-address dynamic-pool-size 5000
```

Configuring IPv6 Inside Clients

Configuring IPv6 Inside Clients by Using the GUI

Configuring IP Lists

1. Navigate to **CGN > Fixed NAT > IP Lists**.
2. Click **Create**.
3. Enter a Name for the list.
4. Select **IPv6**.

NOTE: For DS-Lite, you must select IPv6.

5. To configure IPv6 addresses, click **Add**.
   a. Enter the **Start Address** for the beginning (lowest) IP address and the **End Address** for the ending (highest) IP address in the range.
   b. Click the save icon.

6. To configure IPv6 prefix, click **Add**.
   a. Enter the **Prefix Start**, **Prefix End**, and **Count**.
   b. Click the save icon.
7. Repeat these steps for each IP address range, if applicable.
8. Click **Create**.

Configuring Fixed NAT

To configure Fixed NAT:

1. Navigate to **CGN > Fixed NAT**.
2. Click **Create**.
3. In the **Inside** section, complete the following steps:

NOTE: You can specify one, contiguous IP range or an IP list.

- To specify one IP range:
  a. Select **IP Address**.
  b. Select **IPv4** or **IPv6**.
  c. Enter the Start Address, End Address and Netmask of the range.
    - To specify an IP list:
      - Select **IP List**.
      - Select **IPv4** or **IPv6**.
      - In **IP list**, select the IP list.

4. In **NAT**, complete the following steps:

NOTE: You can specify one, contiguous IP range or an IP list.

- To specify an **IP range**:

a. In the **Inside** section, select **IPv4** or **IPv6**.
b. In **Start Address**, enter the beginning (lowest) IP address in the range.

- To specify an IP list:
  i. In the **Inside** section, select **IPv4** or **IPv6**.
  ii. In the **IP list** drop-down list, select the IP list.

5. Configure general settings, if applicable. Click **General Fields** to display the options. (For more information, see Fixed-NAT Configuration Options.)
6. Click **Create**.

Configuring Fixed-NAT Mappings for DS-Lite by using the CLI

The following commands use IP lists to configure Fixed-NAT for DS-Lite:

1. Enter the following command to configure the IP list for the client IP address ranges:

```
ACOS(config)# ip-list fixed-nat-dslite-users
```

2. Enter the following commands to configures an IP range:

```
ACOS(config-ip list)# 2001::/16 to 200a::/16
ACOS(config-ip list)# ip-list fixed-nat-public-address
ACOS(config-ip list)# 203.0.113.3 to 203.0.113.4
ACOS(config-ip list)# exit
```

3. Enter the following command to configure Fixed-NAT:

```
ACOS(config)# cgnv6 fixed-nat inside ip-list fixed-nat-dslite-users nat
ip-list
fixed-nat-public-address
```

CLI Examples

The following examples help you understand Fixed-NAT.

CLI Example 1

The following command configures Fixed-NAT for DS-Lite CPE with IPv6 addresses in the range 2001:db8::1-100:

```
ACOS(config)# cgnv6 fixed-nat inside 2001:db8::1 2001:db8::100 netmask 64
nat 203.0.113.3 203.0.113.4 netmask /24
```

The following commands show the Fixed-NAT port mappings for the NAT addresses, 203.0.113.3 and 203.0.113.4:

```
ACOS(config)# show cgnv6 fixed-nat nat-address 203.0.113.3 port-mapping
NAT IP Address: 203.0.113.3
Inside User: 2001:db8::1
 TCP:  1024 to 1527
 UDP:  1024 to 1527
 ICMP: 1024 to 1527
Inside User: 2001:db8::2
 TCP:  1528 to 2031
 UDP:  1528 to 2031
 ICMP: 1528 to 2031
Inside User: 2001:db8::3
 TCP:  2032 to 2535
 UDP:  2032 to 2535
 ICMP: 2032 to 2535
Inside User: 2001:db8::4
 TCP:  2536 to 3039
 UDP:  2536 to 3039
 ICMP: 2536 to 3039
Inside User: 2001:db8::5
 TCP:  3040 to 3543
 UDP:  3040 to 3543
 ICMP: 3040 to 3543
...

Inside User: 2001:db8::80
 TCP:  65032 to 65535
 UDP:  65032 to 65535
 ICMP: 65032 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 203.0.113.4 port-mapping
NAT IP Address: 203.0.113.4
Inside User: 2001:db8::81
 TCP:  1024 to 1527
 UDP:  1024 to 1527
 ICMP: 1024 to 1527
Inside User: 2001:db8::82
 TCP:  1528 to 2031
 UDP:  1528 to 2031
```

```
 ICMP: 1528 to 2031
Inside User: 2001:db8::83
 TCP:  2032 to 2535
 UDP:  2032 to 2535
 ICMP: 2032 to 2535
Inside User: 2001:db8::84
 TCP:  2536 to 3039
 UDP:  2536 to 3039
 ICMP: 2536 to 3039
Inside User: 2001:db8::85
 TCP:  3040 to 3543
 UDP:  3040 to 3543
 ICMP: 3040 to 3543
...
Inside User: 2001:db8::100
 TCP:  65032 to 65535
 UDP:  65032 to 65535
 ICMP: 65032 to 65535
```

CLI Example 2

The following commands use IP lists to configure Fixed-NAT for DS-Lite:

```
ACOS(config)# ip-list fixed-nat-dslite-users
ACOS(config-ip list)# 2001::/16 to 200a::/16
ACOS(config-ip list)# ip-list fixed-nat-public-address
ACOS(config-ip list)# 203.0.113.3 to 203.0.113.4
ACOS(config-ip list)# exit
ACOS(config)# cgnv6 fixed-nat inside ip-list fixed-nat-dslite-users nat
ip-list
fixed-nat-public-address
```

The following commands show the Fixed-NAT port mappings for the NAT addresses, 203.0.113.3 and 203.0.113.4:

```
ACOS(config)# show cgnv6 fixed-nat nat-address 203.0.113.3 port-mapping
NAT IP Address: 203.0.113.3
Inside User: 2001::
 TCP:  1024 to 13925
 UDP:  1024 to 13925
 ICMP: 1024 to 13925
```

```
Inside User: 2002::
 TCP:  13926 to 26827
 UDP:  13926 to 26827
 ICMP: 13926 to 26827
Inside User: 2003::
 TCP:  26828 to 39729
 UDP:  26828 to 39729
 ICMP: 26828 to 39729
Inside User: 2004::
 TCP:  39730 to 52631
 UDP:  39730 to 52631
 ICMP: 39730 to 52631
Inside User: 2005::
 TCP:  52632 to 65533
 UDP:  52632 to 65533
 ICMP: 52632 to 65533

ACOS(config)# show cgnv6 fixed-nat nat-address 203.0.113.4 port-mapping
NAT IP Address: 203.0.113.4
Inside User: 2006::
 TCP:  1024 to 13925
 UDP:  1024 to 13925
 ICMP: 1024 to 13925
Inside User: 2007::
 TCP:  13926 to 26827
 UDP:  13926 to 26827
 ICMP: 13926 to 26827
Inside User: 2008::
 TCP:  26828 to 39729
 UDP:  26828 to 39729
 ICMP: 26828 to 39729
Inside User: 2009::
 TCP:  39730 to 52631
 UDP:  39730 to 52631
 ICMP: 39730 to 52631
Inside User: 200a::
 TCP:  52632 to 65533
 UDP:  52632 to 65533
 ICMP: 52632 to 65533
```

Configuring Fixed NAT Address Mapping

The procedures in this section help you choose the method to use for fixed mapping of inside client IP addresses to external NAT addresses.

NOTE: Configuring an address mapping method is completely optional. If you choose not to configure this feature explicitly, the Use Least NAT IPs mapping algorithm will be used.

Configuring the Fixed-NAT Address Mapping Using the GUI

To specify the method to use when mapping inside client IP addresses to the corresponding external NAT IP addresses:

1. Navigate to **CGN > Fixed NAT**.
2. Click **Create**.
3. In the **Inside** section, if you created an IP list for your inside client IP addresses, select IP List. Select the IP list from the drop-down list.
4. If you created an IP list for your external NAT addresses, in **NAT**, select **IP List**.
5. In **IP List**, select a NAT IP List.
6. (Optional) In the General Fields section, for **Method**, select an option.

   If you do not select an option, the default option, Use Least NAT IP Addresses, is used.

7. (Optional) In **Offset**, do one of the following:
   a. Select **Random**.
   b. Select **Value** and enter a value.

      Random allows the ACOS device and software to determine the offset. However, to explicitly configure an offset value, you must enter a value.

8. Click **Create**.

Configuring the Fixed-NAT Address Mapping Using the CLI

You can configure the following address mapping options:

- **use-all-nat-ips** or **use-least-nat-ips** as an address mapping algorithm.
- An offset.

The default offset is 0, but it can be configured as a numeric **value** or as **random**.

NOTE: The offset that you specify should be lesser than the available number of NAT IP addresses.

If you create an IP list for your inside client IP addresses and another for your NAT addresses, you can use this command to specify the address mapping algorithm.

When it comes to specifying offsets, you have two ways in which to configure them:

- Configure the offset explicitly using the **offset** keyword. In this way, you control which inside client IP address will be mapped to a particular NAT IP address of your choice.
- Configure the offset dynamically using the **random** keyword. In this way, the ACOS device and software will automatically assign an offset for the inside client IP address. At the time of configuration, a random offset value will be assigned. If the ACOS device reboots, a different value may be chosen the next time.

Configuration Examples

The following examples assume that you have an IP list for inside client IP addresses called “inside”, as follows:

```
ACOS(config)# ip-list inside
ACOS(config-ip-list)# 12.10.10.160 to 12.10.10.188
```

The examples also assume that you have an IP list for NAT IPs called “nat”, as follows:

```
ACOS(config)# ip-list outside
ACOS(config-ip-list)# 9.9.9.87 to 9.9.9.91
```

The examples display the method that is used when mapping inside addresses to external NAT IP addresses. When you configure the Fixed-NAT address mapping method, the show commands display the five insider users who are mapped to each NAT IP address and their corresponding port allocations.

Using Least NAT IP Addresses

The following example displays the Use Least NAT IP addresses method that is used when mapping inside addresses to external NAT IP addresses. Note that the first inside client IP address is mapped to the first NAT Address:

```
ACOS(config)# cgnv6 fixed-nat inside ip-list inside nat ip-list outside
method use-least-nat-ips

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.87 port-mapping
NAT IP Address: 9.9.9.87
Inside User: 12.10.10.160
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.161
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.162
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.163
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.164
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.165
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.88 port-mapping
NAT IP Address: 9.9.9.88
Inside User: 12.10.10.166
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.167
```

```
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.168
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.169
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.170
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.171
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.89 port-mapping
NAT IP Address: 9.9.9.89
Inside User: 12.10.10.172
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.173
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.174
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.175
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.176
 TCP:  44032 to 54783
```

```
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.177
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.90 port-mapping
NAT IP Address: 9.9.9.90
Inside User: 12.10.10.178
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.179
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.180
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.181
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.182
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.183
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.91 port-mapping
NAT IP Address: 9.9.9.91
Inside User: 12.10.10.184
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
```

```
Inside User: 12.10.10.185
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.186
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.187
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.188
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
```

Using Least NAT IP Addresses with an Offset

This example configures the Use Least NAT IPs method with an offset of 2. Note that the first inside client IP address is mapped to the third NAT address based on the offset of 2 that you specified. When all the inside clients are mapped once, the cycle continues from the first IP address again.

```
ACOS(config)# cgnv6 fixed-nat inside ip-list inside nat ip-list outside
method use-least-nat-ips offset 2
ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.89 port-mapping
NAT IP Address: 9.9.9.89
Inside User: 12.10.10.160
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.161
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.162
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.163
```

```
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.164
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.165
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.90 port-mapping
NAT IP Address: 9.9.9.90
Inside User: 12.10.10.166
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.167
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.168
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.169
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.170
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.171
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.91 port-mapping
```

```
NAT IP Address: 9.9.9.91
Inside User: 12.10.10.172
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.173
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.174
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.175
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.176
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.177
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.87 port-mapping
NAT IP Address: 9.9.9.87
Inside User: 12.10.10.178
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.179
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.180
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
```

```
Inside User: 12.10.10.181
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.182
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.183
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.88 port-mapping
NAT IP Address: 9.9.9.88
Inside User: 12.10.10.184
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.185
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.186
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.187
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.188
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
```

Using Least NAT IP Addresses with Offset Random

This example configures the Use Least NAT IPs method with a random offset assigned by ACOS.

```
ACOS(config)# cgnv6 fixed-nat inside ip-list inside nat ip-list outside
method use-least-nat-ips offset random
ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.91 port-mapping
NAT IP Address: 9.9.9.91
Inside User: 12.10.10.160
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.161
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.162
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.163
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.164
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.165
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.87 port-mapping
NAT IP Address: 9.9.9.87
Inside User: 12.10.10.166
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.167
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.168
```

```
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.169
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.170
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.171
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.88 port-mapping
NAT IP Address: 9.9.9.88
Inside User: 12.10.10.172
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.173
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.174
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.175
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.176
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.177
 TCP:  54784 to 65535
```

```
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.89 port-mapping
NAT IP Address: 9.9.9.89
Inside User: 12.10.10.178
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.179
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.180
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.181
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.182
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.183
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.90 port-mapping
NAT IP Address: 9.9.9.90
Inside User: 12.10.10.184
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.185
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
```

```
Inside User: 12.10.10.186
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.187
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.188
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
```

Using All NAT IP Addresses

The following example displays the Use All NAT IPs method that is used when mapping inside client IP addresses to external NAT IP addresses. Note that the first inside client IP address is mapped to the first NAT address, the second client IP address to the second NAT IP address, and so on.

When all the first five client IP addresses (from the client IP address range of 12.10.10.160-188) are assigned to the five available NAT IP addresses (9.9.9.87-91), one per NAT IP address, the cycle continues with the sixth client IP address (12.10.10.165) being mapped to the first NAT IP address (9.9.9.87). This mapping continues until all client IP addresses are assigned an external NAT IP address. This mapping model ensures that no NAT IP address is left unused:

```
ACOS(config)# cgnv6 fixed-nat inside ip-list inside nat ip-list outside
method use-all-nat-ips

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.87 port-mapping
NAT IP Address: 9.9.9.87
Inside User: 12.10.10.160
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.165
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.170
 TCP:  22528 to 33279
```

```
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.175
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.180
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.185
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535
ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.88 port-mapping
NAT IP Address: 9.9.9.88
Inside User: 12.10.10.161
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.166
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.171
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.176
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.181
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.186
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535
```

```
ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.89 port-mapping
NAT IP Address: 9.9.9.89
Inside User: 12.10.10.162
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.167
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.172
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.177
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.182
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.187
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.90 port-mapping
NAT IP Address: 9.9.9.90
Inside User: 12.10.10.163
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.168
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.173
 TCP:  22528 to 33279
```

```
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.178
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.183
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.188
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.91 port-mapping
NAT IP Address: 9.9.9.91
Inside User: 12.10.10.164
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.169
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.174
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.179
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.184
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
```

Using All NAT IPs with an Offset

The following example displays the Use All NAT IPs method with an offset of 2. Note that the first inside client IP address is mapped to the third NAT address based on the offset of 2 that you specified. When all the inside clients are mapped once, the cycle continues from the first IP address again.

The following example displays the effect of the configured offset of 2, starting with the NAT IP Address that contains the first inside client IP address of 12.10.10.160:

```
ACOS(config)# cgnv6 fixed-nat inside ip-list inside nat ip-list outside
method use-all-nat-ips offset 2

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.89 port-mapping
NAT IP Address: 9.9.9.89
Inside User: 12.10.10.160
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.165
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.170
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.175
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.180
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.185
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.90 port-mapping
NAT IP Address: 9.9.9.90
Inside User: 12.10.10.161
```

```
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.166
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.171
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.176
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.181
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.186
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.91 port-mapping
NAT IP Address: 9.9.9.91
Inside User: 12.10.10.162
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.167
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.172
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.177
 TCP:  33280 to 44031
```

```
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.182
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.187
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.87 port-mapping
NAT IP Address: 9.9.9.87
Inside User: 12.10.10.163
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.168
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.173
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.178
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.183
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.188
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.88 port-mapping
NAT IP Address: 9.9.9.88
```

```
Inside User: 12.10.10.164
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.169
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.174
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.179
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.184
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
```

Use All NAT IPs with Offset Random

The following example displays the Use All NAT IPs method with a random offset.

```
ACOS(config)# cgnv6 fixed-nat inside ip-list inside nat ip-list outside
method use-all-nat-ips offset random
ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.88 port-mapping
NAT IP Address: 9.9.9.88
Inside User: 12.10.10.160
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.165
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.170
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.175
```

```
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.180
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.185
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.89 port-mapping
NAT IP Address: 9.9.9.89
Inside User: 12.10.10.161
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.166
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.171
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.176
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.181
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.186
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.90 port-mapping
```

```
NAT IP Address: 9.9.9.90
Inside User: 12.10.10.162
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.167
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.172
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.177
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.182
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.187
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.91 port-mapping
NAT IP Address: 9.9.9.91
Inside User: 12.10.10.163
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.168
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.173
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
```

```
Inside User: 12.10.10.178
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.183
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
Inside User: 12.10.10.188
 TCP:  54784 to 65535
 UDP:  54784 to 65535
 ICMP: 54784 to 65535

ACOS(config)# show cgnv6 fixed-nat nat-address 9.9.9.87 port-mapping
NAT IP Address: 9.9.9.87
Inside User: 12.10.10.164
 TCP:  1024 to 11775
 UDP:  1024 to 11775
 ICMP: 1024 to 11775
Inside User: 12.10.10.169
 TCP:  11776 to 22527
 UDP:  11776 to 22527
 ICMP: 11776 to 22527
Inside User: 12.10.10.174
 TCP:  22528 to 33279
 UDP:  22528 to 33279
 ICMP: 22528 to 33279
Inside User: 12.10.10.179
 TCP:  33280 to 44031
 UDP:  33280 to 44031
 ICMP: 33280 to 44031
Inside User: 12.10.10.184
 TCP:  44032 to 54783
 UDP:  44032 to 54783
 ICMP: 44032 to 54783
```

Configuring MAC-based Nexthop Routing for Fixed-NAT

MAC-based nexthop routing for Fixed-NAT enables the ACOS device to identify the route hop based on the MAC address of the inside client's request. The ACOS device uses the MAC address, instead of the route table, to select the next hop for the reply. Replies that are sent to the client use the same route hop on which the request was received.

Consider the following information:

- MAC-based nexthop routing is supported for LSN, DS-Lite, NAT64 and Fixed-NAT sessions and is not supported for Stateless NAT. You can use MAC-based nexthop routing with 6rd, only if 6rd is used with NAT64.
- This enhancement operates on a per-session basis. After the session idles out, the client's MAC address is no longer used to identify the route hop and the ACOS device looks at the route table to send the reply.

Enabling MAC-based Nexthop Routing for Fixed-NAT Using the GUI

To enable MAC-based nexthop routing for Fixed-NAT by using the GUI:

1. Navigate to **CGN > LSN > LSN LID**.
2. Click **Create**.
3. Select the **Respond to User MAC** check box.
4. (Optional) Modify the other configurations, if necessary.
5. Click **Create**.

Enabling MAC-based Nexthop Routing for Fixed-NAT Using the CLI

1. Enter the following command to enter the configuration level for an LSN LID:

```
ACOS(config)# cgnv6 lsn-lid 22
```

2. Enter the following command to enable MAC-based nexthop routing for the specified LSN LID:

```
ACOS(config-lsn-lid)# respond-to-user-mac
```

Configuring L3V Inter-partition Routing for Fixed-NAT

Configuring Fixed-NAT in a L3V Deployment Using CLI

The commands in this section configure Fixed-NAT for two L3V partitions, “p2” and “p3”.

Shared Partition Configuration

The following commands configure an Ethernet interface to the Internet and enable NAT outside on the interface:

```
ACOS(config)# vlan 20
ACOS(config-vlan:20)# tagged ethernet 2
ACOS(config-vlan:20)# router-interface ve 20
ACOS(config-vlan:20)# interface ve 20
ACOS(config-if:ve20)# ip address 9.9.10.200 255.255.255.0
ACOS(config-if:ve20)#ip nat outside
ACOS(config-if:ve20)# exit
```

The following commands configure VRRP-A for the shared partition:

```
ACOS(config)# vrrp-a common
ACOS(config-common)# device-id 1
ACOS(config-common)# set-id 1
ACOS(config-common)# enable
ACOS(config-common)# exit
ACOS-Active(config)# vrrp-a vrid l
ACOS-Active(config-vrid:1)# vrrp-a vrid-lead vrid1-leader
ACOS-Active(config-vrid-lead:vrid1-leader)# partition shared vrid 1
ACOS-Active(config-vrid-lead:vrid1-leader)# exit
```

The **`vrrp-a vrid-lead`** option configures a VRRP-A lead. Later in the configuration, the partitions are configured to follow the shared partition’s VRID state. When the shared partition’s VRID is active, so is the VRID of each of the private partitions that follows the shared partition’s VRID state. Likewise, if the shared partition’s VRID state changes to Standby, so does the VRID state of each of the private partition VRIDs that are followers.

The following commands create the L3V partitions:

```
ACOS-Active(config)# partition p0 id 10 application-type cgnv6
```

```
ACOS-Active(config-partition:p0)# exit
ACOS-Active(config)# partition p1 id 20 application-type cgnv6
ACOS-Active(config-partition:p1)# exit
```

The following commands configure IP lists for use by partition “p2”:

```
ACOS-Active(config)# ip-list P2_inside_users
ACOS-Active(config-ip list)# 10.10.10.1 to 10.10.10.100
ACOS-Active(config-ip list)# 192.168.210.1 to 192.168.210.100
ACOS-Active(config-ip list)# ip-list P2_NAT-IPs
ACOS-Active(config-ip list)# 9.9.9.1 to 9.9.9.10
```

The following commands configure IP lists for use by partition “p3”:

```
ACOS-Active(config-ip list)# ip-list P3_inside_users
ACOS-Active(config-ip list)# 10.10.10.1 to 10.10.10.100
ACOS-Active(config-ip list)# 192.168.210.1 to 192.168.210.100
ACOS-Active(config-ip list)# ip-list P3_NAT-IPs
ACOS-Active(config-ip list)# 9.9.9.11 to 9.9.9.20
```

The following command configures Fixed-NAT mappings for partition “p2” making use of the IP lists that were just created:

```
ACOS(config-ip list)# cgnv6 fixed-nat inside ip-list P2_inside_users
partition p2 nat ip-list P2_NAT_IPs vrid 1
```

For partition “p3”:

```
ACOS(config)# cgnv6 fixed-nat inside ip-list P3_inside_users partition p3
nat ip-list P3_NAT_IPs vrid 1
```

The partition option in the Fixed-NAT configuration above binds them for use by a specific partition alone.

NOTE: The NAT addresses need to be unique in the case of inter-partition routing for Fixed-NAT. Inside client IP addresses can overlap across different private partitions. Hairpinning is supported between inside clients belonging to different private partitions (even with overlapping IP addresses).

Private Partition p2 Configuration

The commands in this section are used to configure the private partition “p2”. To begin, the following command changes the CLI to the partition “p2”.

```
ACOS-Active(config)# end
ACOS-Active# active-partition p2
Currently active partition: p2
```

First, configure an Ethernet interface to the inside clients and enable NAT inside on the interface:

```
ACOS-Active[p2]# configure
ACOS-Active[p2](config)# vlan 10
ACOS-Active[p2](config-vlan:10)# tagged ethernet 1
ACOS-Active[p2](config-vlan:10)# router-interface ve 10
ACOS-Active[p2](config-if:ve10)# interface ve 10
ACOS-Active[p2](config-if:ve10)# ip address 10.10.10.200 255.255.255.0
ACOS-Active[p2](config-if:ve10)# ip nat inside
ACOS-Active[p2](config-if:ve10)# exit
```

The following command configures a static IP route from the private partition to the shared partition. The **partition shared** option specifies that the next hop for the route is the shared partition.

```
ACOS-Active[p2](config)# ip route 0.0.0.0 /0 partition shared
```

The following command enables VRRP-A for the partition. VRID 1 is configured to base its Active/Standby state on the state of VRID lead “leader”, configured in the shared partition:

```
ACOS-Active[p2](config)# vrrp-a vrid 1
ACOS-Active[p2](config-vrid:1)# follow vrid-lead vrid1-leader
```

Private Partition p3 Configuration

The commands in this section mirror those used to configure partition “p3”.

NOTE: VLANs and VEs need to be unique across partitions but interface addresses do not need to be unique.

```
ACOS-Active(config)# end
ACOS-Active# active-partition p3
```

```
Currently active partition: p3
ACOS-Active[p3]# configure
ACOS-Active[p3](config)# vlan 11
ACOS-Active[p3](config-vlan:11)# tagged ethernet 1
ACOS-Active[p3](config-vlan:11)# router-interface ve 11
ACOS-Active[p3](config-if:ve11)# interface ve 11
ACOS-Active[p3](config-if:ve11)# ip address 10.10.10.200 255.255.255.0
ACOS-Active[p3](config-if:ve11)# ip nat inside
ACOS-Active[p3](config-if:ve11)# exit
ACOS-Active[p3](config)# vrrp-a vrid 1
ACOS-Active[p3](config-vrid:1)# follow vrid-lead vrid1-leader
```

Enhanced Fixed-NAT Table Accessibility

Up to ten port-mapping files can exist for a Fixed-NAT configuration. When the maximum configured number of configuration files is exceeded, then the oldest file will be deleted in order to add the new configuration file. It is also possible to manually delete a Fixed-NAT configuration file, although the newest file with the current configuration cannot be deleted. Only past configuration files can be deleted.

If file creation for port mapping is enabled, a new file is created whenever the ACOS device reloads, and a syslog entry will be created to indicate the new file name. A notification is issued when a new file is created via an SNMP trap. Likewise, whenever a file is deleted, an SNMP trap and a syslog message are generated.

For the new Fixed-NAT configuration files, there will be a timestamp to indicate the time of file creation. When the Fixed-NAT Table information is exported, an MD5 checksum is included at the end of the file to indicate whether or not the transfer of Fixed-NAT table information is complete.

When a Fixed-NAT configuration is deleted, then the latest file will be saved in an archive. An appended timestamp on the file will indicate the time the configuration was deleted.

To access the archived file, use a show command or export the file.

Configuring Exported Fixed-NAT Table Information

To enable Fixed-NAT port mapping files, enter the following command at the global configuration level:

```
ACOS(config)# cgnv6 fixed-nat create-port-mapping-file
```

To configure the number of port-mapping files to retain, enter the following command at the global configuration level:

```
ACOS(config)# cgnv6 fixed-nat port-mapping-files-count 1
```

The number of stored port-mapping files can range from 1 to 10. The default value is 5 files.

To delete a port-mapping file manually, enter the following command at the global configuration level:

```
ACOS(config)# delete cgnv6 fixed-nat file1
```

To display a list of active port-mapping files, reflecting the current configuration, enter the following command:

```
ACOS(config)# show cgnv6 fixed-nat port-mapping-file
```

To display a list of all port-mapping files, including past configurations, enter the following command:

```
ACOS(config)# show cgnv6 fixed-nat port-mapping-files all
```

To display the archived Fixed-NAT port-mapping file, enter the following command:

```
ACOS(config)# show cgnv6 fixed-nat port-mapping-files archive
```

To export the archived Fixed-NAT port-mapping file, enter the following command:

```
ACOS(config)# export fixed-nat-archive file1
ftp://user:single@192.168.219.234/a.txt
```

To periodically back-up Fixed-NAT files, enter the following command at the global configuration level:

```
ACOS(config)# backup-periodic fixed-nat  {hour num | day num |weeks num}
```

The num for periodic backup specifies the hour, day, or weekly interval in which to back-up the Fixed-NAT files. The hourly interval can be from 1 to 65534 hours. The daily interval can be from 1 to 199 days. The weekly interval can be from 1 to 199 hours.

NOTE: The periodic backup applies only to the active-port mapping files that reflect the current configuration. This backup does not export the historical files.

SNMP for Fixed-NAT Table Information

When a new Fixed-NAT configuration file is created, the following SNMP trap is generated, with the corresponding OID:

NOTE: The SNMP trap is generated while the file is created, and the trap contains only the filename and the event of trigger (Creation, Deletion).

The format of the filename is:
fixed_nat_[<NAT_IP_START>_<NAT_IP_END>][ip_list_IPLIST_NAME]_<TIME_STAMP>.

Here is an example:

```
fixed_nat_217.7.7.7_217.7.7.8_2014_12_03_165228. The time stamp in this
case is December 03 2014 at 16 hours 33 minutes and 28 seconds.
axLsnFixedNatPortMappingFileChange
1.3.6.1.4.1.22610.2.4.3.12.2.4.14
```

To get the list of all of the active files, as well as the timestamp information, use the following SNMP method and OID to do a status check for configuration changes:

```
axFixedNatFileTable
1.3.6.1.4.1.22610.2.4.3.18.120.16.1
```

This method has two elements:

- axFixedNatFileName
- axFixedNatFileTimeStamp

Exporting Fixed-NAT Table Information Using aXAPI

The aXAPI can be used to delete, download or view Fixed NAT mapping files.

lsn.fixed_nat.port_mapping_file.getAll

This method retrieves a list of all port mapping files, and their names. Optionally, use the input parameter only_active to specify whether to retrieve only active files (1), or

to retrieve all files (0). By default, only active files are retrieved.

Example

This example shows how to use the aXAPI to retrieve all fixed NAT tables.

URL:

```
https://[IP]:[Port]/services/rest/V2.8/?session_id=[SESSION_
ID]&format=json&method=lsn.fixed_nat.port_mapping_file.getAll
```

HTTP POST Body:

```
{
        “only_active”: 0
}
```

aXAPI Response:

```
{
        "fixed_nat_port_mapping_file_list":[
                {
                        "name":"fixed_nat_3.3.3.3_2014_11_11_072253"
                },
                {
                        "name":"fixed_nat_3.3.3.3_2014_11_06_045122"
                },
                {
                        "name":"fixed_nat_3.3.3.3_2014_11_06_025145"
                },
                {
                        "name":"fixed_nat_69.9.9.9_2014_11_11_072253"
                },
                {
                        "name":"fixed_nat_69.9.9.9_2014_11_06_081140"
                },
                {
                        "name":"fixed_nat_69.9.9.9_2014_11_06_070759"
                },
                {
                        "name":"fixed_nat_22.2.2.2_2014_11_11_072253"
                },
                {
```

```
                        "name":"fixed_nat_22.2.2.2_2014_11_06_081140"
                },
                {
                        "name":"fixed_nat_22.2.2.2_2014_11_06_070924"
                }
        ]
}
```

Example

This example shows how to use the aXAPI to retrieve all currently active fixed NAT tables.

URL:

```
http(s)://[IP]:[Port]/services/rest/V2.8/?session_id=[SESSION_
ID]&format=json&method=lsn.fixed_nat.port_mapping_file.getAll
```

HTTP POST Body: (Optional, because 1 is the default value of the only_active parameter)

```
{
        “only_active”: 1
}
```

aXAPI Response:

```
{
        "fixed_nat_port_mapping_file_list":[
                {
                        "name":"fixed_nat_3.3.3.3_2014_11_11_072253"
                },
                {
                        "name":"fixed_nat_69.9.9.9_2014_11_11_072253"
                },
                {
                        "name":"fixed_nat_22.2.2.2_2014_11_11_072253"
                }
        ]
}
```

lsn.fixed_nat.port_mapping_file.delete

This method deletes the named file. Specify the file name using the input parameter name, a string of 1-255 characters.

Example

This example deletes the file “fixed_nat_22.2.2.2_2014_11_06_070924”.

URL:

```
http(s)://[IP]:[Port]/services/rest/V2.8/?session_id=[SESSION_
ID]&format=json&method=lsn.fixed_nat.port_mapping_file.delete
```

HTTP POST Body:

```
{
        "name": "fixed_nat_22.2.2.2_2014_11_06_070924"
}
```

lsn.fixed_nat.port_mapping_file.download

This method downloads the named file. Specify the file name using the input parameter name, a string of 1-255 characters.

Do not download using the POST body. Use the URL for the input parameter instead.

Example

This example shows how to use CURL to download a fixed NAT file as a text file.

URL:

```
http(s)://[IP]:[Port]/services/rest/V2.8/?session_id=[SESSION_
ID]&format=json&method=lsn.fixed_nat.port_mapping_
file.download&name=fixed_nat_22.2.2.2_2014_11_06_070924
```

CURL Command:

Use CURL to download the fixed NAT port-mapping file “fixed_nat_6.6.6.6_2014_11_11_072254” to the file “fixed_nat_down.txt”

```
curl -o fixed_nat_down.txt  -k http:// [IP]:[Port]
/services/rest/V2.8/?session_id=[SESSION_ID] \&method=lsn.fixed_nat.port_
mapping_file.download\&name=fixed_nat_6.6.6.6_2014_11_11_072254
```

Displaying Fixed-NAT Information

Displaying Fixed-NAT Port Mappings

Enter the following commands to display Fixed-NAT port mappings for a specific inside client IP address:

```
ACOS(config)# show cgnv6 fixed-nat inside-user 1.1.1.1 port-mapping
NAT IP Address: 2.1.1.1
 TCP:  1024 to 65535
 UDP:  1024 to 65535
 ICMP: 1024 to 65535
```

Enter the following commands to display Fixed-NAT port mappings for a NAT address:

```
ACOS(config)# show cgnv6 fixed-nat nat-address 2.1.1.5 port-mapping
NAT IP Address: 2.1.1.5
Inside User: 1.1.1.5
 TCP:  1024 to 65535
 UDP:  1024 to 65535
 ICMP: 1024 to 65535
```

Displaying Current Port and Session Use for a Fixed-NAT Client

Enter the following command to list the number of sessions a client currently has active, and the number of TCP, UDP, and ICMP ports in use by the client:

```
AX3000(config)# show cgnv6 fixed-nat inside-user 1.1.1.1 quota-used
NAT IP Address: 2.1.1.1
Session Quota Used: 0
TCP Ports Used:       0
UDP Ports Used:       0
ICMP Resources Used: 0
```

Displaying the Full-cone Sessions for a Fixed-NAT NAT Address

Enter the following command to display the full-cone sessions that are using a specific NAT address:

```
ACOS(config)# show cgnv6 fixed-nat full-cone-sessions nat-address ipv4addr
```

Displaying Fixed-NAT Statistics

Enter the following command to display statistics for Fixed-NAT:

```
ACOS(config)# show cgnv6 fixed-nat statistics
```

Enter the following command to show a histogram of active TCP or UDP users using ports in specific port ranges:

```
ACOS(config)# show cgnv6 fixed-nat histogram port-usage {inside-user | nat-ip}
```

For more information on show commands, see *Command Line Interface*.

Removing Fixed-NAT Configuration

Before removing a Fixed-NAT configuration, it is recommended to first disable Fixed-NAT, clear the associated sessions, wait for all the active users to be cleared, and then remove the configuration.

If a Fixed-NAT configuration is removed when there are active sessions still using the NAT IP addresses, there is a chance that a NAT IP address can be reused before the old session is cleared. This can result in unknown security issues. To prevent such issues, a warning message is displayed. It is recommended to wait for approximately 2 minutes for all the associated sessions to be cleared before deleting the Fixed-NAT configuration.

To remove a Fixed-NAT configuration gracefully, perform the following:

- Disabling a Fixed-NAT Configuration
- Deleting a Fixed-NAT Configuration

To reconfigure a Fixed-NAT configuration and reuse the NAT IP address, see Reconfiguring a Fixed-NAT Configuration and Reusing NAT IP Address.

Disabling a Fixed-NAT Configuration

To disable a Fixed-NAT configuration and clear the active sessions, perform the following:

Using GUI

1. Navigate to **CGN > Fixed NAT**.
2. Select the Fixed NAT configuration you wish to remove.
3. Click **Disable**.

   The following message is displayed, “Do you want to disable fixed-nat configuration?”.

4. Click **Yes**.

   When the Fixed-NAT configuration is disabled, the new sessions are stopped and the existing sessions using the configuration are cleared.

To view the disabled Fixed-NAT configuration, navigate to **CGN > Disabled Fixed NAT**.

Table 17describes the details about the disabled Fixed NAT configurations.

Table 17 : Disabled Fixed NAT

Field	Description
Inside Address	Displays the inside IP address of the Fixed NAT mapping.
Inside IP List	Displays the name of the Inside IP List
Partition	Displays the partition to which the configuration applies.
Active Users	Displays the number of active users using the configuration.
Clear Session	Indicates whether the sessions are cleared or still exists. If Clear Session is displayed as 1, it means that the

Table 17 : Disabled Fixed NAT

Field	Description
	sessions are being cleared in the background. If Clear Session is displayed as 0, it means that the Fixed-NAT configuration is only disabled. The sessions are not cleared.

Using CLI

To disable the Fixed-NAT configuration and clear the active sessions, use the following example:

```
ACOS(config)#cgnv6 fixed-nat disable 12.10.10.163 12.10.10.163 netmask /24
clear-session
```

Clearing active sessions is optional. It can be executed only if you want to remove the Fixed-NAT configuration with minimal downtime. Otherwise, the configuration can be removed after all the existing sessions end.

Deleting a Fixed-NAT Configuration

Before deleting a Fixed-NAT configuration, it is recommended to wait for all sessions using the configuration to be cleared.

To check if the sessions are cleared, you can perform one of the following:

- Run the `show log` command to display the log message.

  For Example,

  ```
  Mar 22 2019 09:58:54 Notice [ACOS]:Fixed NAT configuration for inside
  12.10.10.163 12.10.10.163 netmask /24 can be deleted now.
  Mar 22 2019 09:58:21 Notice [ACOS]:Fixed NAT configuration for inside
  12.10.10.163 12.10.10.163 netmask /24 was disabled for removing.
  Force clear session: True
  ```

- Run the `show cgnv6 fixed-nat nat-address [IP address] disabled-config` command

  For Example,

```
ACOS(config)#show cgnv6 fixed-nat 12.10.10.163 12.10.10.163 netmask /24
disabled-config
Disabled fixed-nat configuration:
==================================
Inside address: 12.10.10.163 12.10.10.163 /24
Inside ip-list:
Inside Partition:
Active users:   0
Clear session: 0
```

Once you verify that the sessions are cleared and active users is 0, you can delete the Fixed-NAT configuration.

Using GUI

1. Navigate to **CGN > Fixed NAT**.
2. Select the Fixed NAT configuration you wish to delete.
3. Click **Delete**.

   The following message is displayed, “Are you sure to delete the selected items?”.
4. Click **Yes**.

Using CLI

To delete the Fixed-NAT configuration, use the following command:

```
ACOS(config)# no cgnv6 fixed-nat inside 12.10.10.163 12.10.10.163 netmask
/24
```

Reconfiguring a Fixed-NAT Configuration and Reusing NAT IP Address

The example in this section illustrates the steps to reconfigure Fixed-NAT and reuse the NAT IP address.

1. Identify the vrid of the standby box for which you want to change the Fixed-NAT configuration.

For example,

```
cgnv6 fixed-nat inside 12.10.10.163 12.10.10.163 netmask /24 nat
9.9.9.91 netmask /24 vrid1
```

Run the `show vrrp-a` command.

```
ACOS-132-Standby(config)#show vrrp-a vrid 1
Unit            State           Weight          Priority
2 (Local)       Standby         65534           100
 *
                became  Standby at:  Mar 22 08:38:33 2019
                          for  0 Day, 0 Hour,28 min
1 (Peer)        Active          65534           150
vrid that is running: 1
```

You must use the standby box to change the Fixed-NAT configuration.

2. On the standby box, disable the Fixed-NAT configuration. You can use the `clear-session` command to clear the active sessions and remove the configuration with minimal downtime.

   Use the following command to disable and clear the sessions:

   ```
   ACOS(config)# cgnv6 fixed-nat disable 12.10.10.163 12.10.10.163 netmask
   /24 clear-session
   ```

3. Delete the Fixed-NAT configuration.

   Check if the active sessions are cleared before deleting the configuration. Perform the following:

   a. Run the `show log` command to display the log message.

      For Example,

      ```
      Mar 22 2019 09:58:54 Notice [ACOS]:Fixed NAT configuration for
      inside  12.10.10.163 12.10.10.163 netmask /24 can be deleted now.
      Mar 22 2019 09:58:21 Notice [ACOS]:Fixed NAT configuration for
      inside  12.10.10.163 12.10.10.163 netmask /24 was disabled for
      removing.
      Force clear session: True
      ```

   b. Run the `show cgnv6 fixed-nat nat-address [IP address] disabled-`

`config` command to view the disabled Fixed-NAT configurations.

```
ACOS(config)#show cgnv6 fixed-nat 12.10.10.163 12.10.10.163 netmask
/24 disabled-config
Disabled fixed-nat configuration:
=================================
Inside address: 12.10.10.163 12.10.10.163 /24
Inside ip-list:
Inside Partition:
Active users:   0
Clear session: 0
```

After verifying that the sessions are cleared and active users is 0, run the `no cgnv6 fixed-nat` command to delete the configuration.

```
ACOS(config)# no cgnv6 fixed-nat inside 12.10.10.163 12.10.10.163
netmask /24 nat 9.9.9.91 netmask /24 vrid1
```

4. Swap the active and standby box on the configured vrid to add th new Fixed-NAT configuration.

   Log into the active box and modify the priority to change the box to standby.

```
ACOS-Active(config)#vrrp-a vrid 1
ACOS-Active(config-vrid:1)#blade-parameters
ACOS-Active(config-vrid:1-blade-parameters)#priority 1
```

   At this point, the standby box is promoted to be the new active box.

```
ACOS-Standby(config-vrid:1-blade-parameters)#show vrrp-a
vrid 1
Unit            State           Weight          Priority
1 (Local)       Standby         65534           1
 *
                became  Standby at:  Mar 17 06:14:12 2019
                          for  0 Day, 0 Hour, 1 min
2 (Peer)        Active          65534           100
              vrid that is running: 1
```

5. On the current standby box, disable the old Fixed-NAT configuration and clear the associated sessions.

```
ACOS-Standby(config)# cgnv6 fixed-nat disable 12.10.10.163 12.10.10.163
netmask /24 clear-session
```

6. On the current active box, configure the new Fixed-NAT configuration. To minimize the downtime, you can configure a new Fixed-NAT configuration soon after step 5. You do not have to wait for all the sessions to be cleared.

```
ACOS-Active(config)# cgnv6 fixed-nat inside 12.10.10.163 12.10.10.164
netmask /24 nat 9.9.9.91 9.9.9.92 netmask /24 vrid 1
```

7. Log into the current standby box, delete the old configuration, and configure the new Fixed-NAT. It is important to make sure the old sessions are cleared using the `show log` and `show cgnv6 fixed-nat nat-address [IP address] disabled-config` commands.

```
ACOS-Standby(config)#no cgnv6 fixed-nat inside 12.10.10.163
12.10.10.163 netmask /24 nat 9.9.9.91 9.9.9.91 netmask /24 vrid 1
ACOS-Standby(config)#cgnv6 fixed-nat inside 12.10.10.163 12.10.10.164
netmask /24 nat 9.9.9.91 9.9.9.92 netmask /24 vrid 1
```

8. Make sure to write memory to save configuration.

Lightweight 4over6

This chapter describes what Lightweight 4over6 is and how to configure it.

The following topics are covered:

Overview

Lightweight 4over6 enables the ACOS device to route traffic between an IPv4 client's IPv6 Customer Premises Equipment (CPE) and IPv4 servers.

In a Lightweight 4over6 deployment, the IPv4 client's CPE performs NAT to assign a public IPv4 address to the client and encapsulates the client's NATed IPv4 traffic in an IPv6 tunnel that is terminated on the ACOS device. ACOS supports multiple tunnel-endpoint addresses in the binding table. For details, see Binding Table.

NOTE: For tunneled packets, the Customer Edge (CE) router decrements the Time to Live (TTL) for the inner packet and drops the packet if TTL is exceeded prior to encapsulation in the tunnel. The device acts as the Border Relay (BR) and terminates the tunnel and will not decrement the TTL for the inner packet.

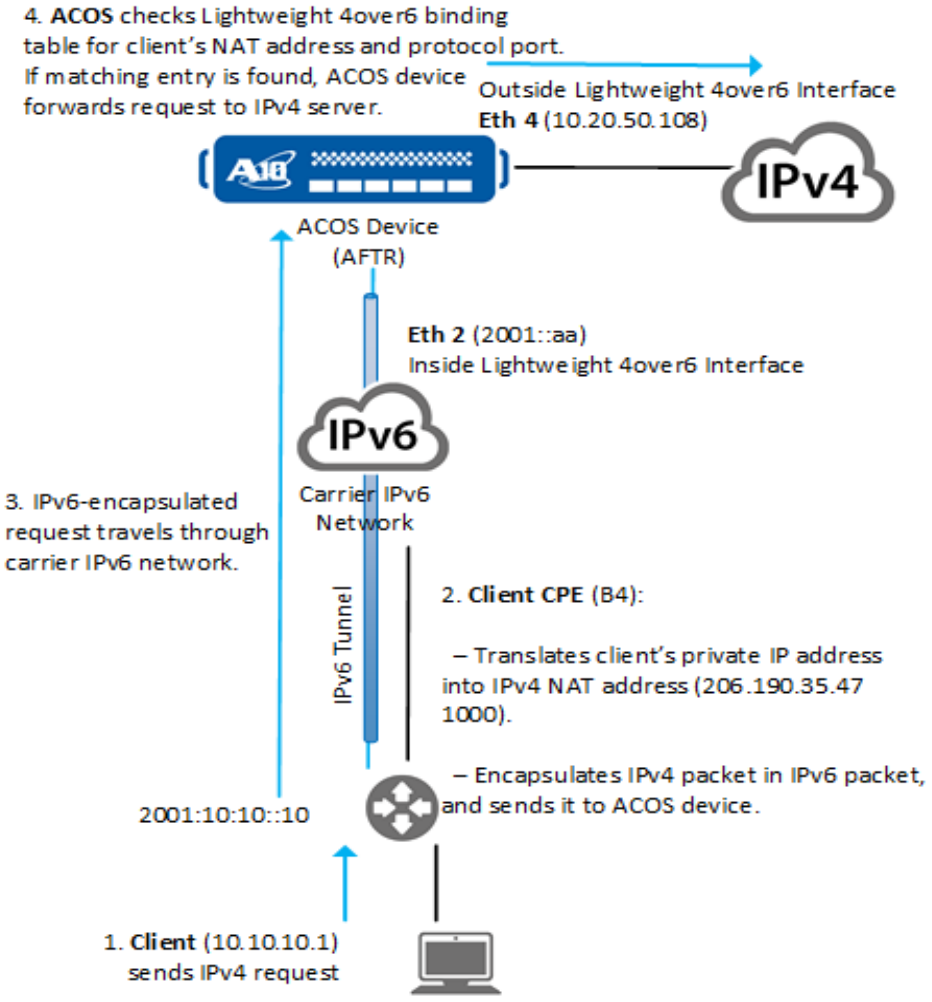
The implementation of this feature is based on the Lightweight 4over6: An Extension to the DS-Lite Architecture, draft-cui-softwire-b5-translated-ds-lite-07 RFC.

In the RFC terminology, Lightweight 4over6 moves the IPv4 NAT function from the *AFTR* device to the *B4* device. The *AFTR* device is the ACOS device, and the *B4* device is the client CPE.

Lightweight 4over6 is supported in the shared partition and L3V private partitions.

Figure 30 shows an example of a Lightweight 4over6 deployment.

Figure 30 : Lightweight 4over6



In the following example, an IPv6 client has the following resources:

- Private IPv4 address 10.10.10.1.
- CPE address 2001:10:10::10. This is the client's IPv6 tunnel address.
- NAT (public) IPv4 address 206.190.35.47:1000.

When the IPv4 client (10.10.10.1) sends an IPv4 request, the client's CPE completes the following tasks:

- Translates the source IP address of the request into a NAT address (206.190.35.47:1000 in this example)

- Encapsulates the entire IPv4 datagram of the request in an IPv6 packet
- Sends the encapsulated request to the ACOS device, over the IPv6 tunnel

ACOS checks the active Lightweight 4over6 binding table. For more information about the binding table, see Binding Table. If the traffic matches a binding-table entry, the ACOS device decapsulates the client request and sends it to the IPv4 server.

Binding Table

ACOS uses a Lightweight 4over6 binding table to recognize valid Lightweight 4over6 traffic. Each entry in the binding table consists of the following items:

- IPv6 address of the client CPE – This is the IPv6 address of the remote end of the tunnel.
- IPv4 NAT address the CPE assigns to the client – This must be a host address, not a subnet address.
- Protocol port number or range – This is the range of ports the CPE may use as source ports with the IPv4 NAT address assigned to the client by the CPE.
- Tunnel endpoint address.

You can configure the binding table on the ACOS device, or import a binding table configured on another device such as a laptop.

A binding table can contain up to 4 million entries, and up to 10 binding-table files can be stored on each partition. Only one binding table can be active at any given time.

NOTE: DS-Lite can be used for traffic that does not match the binding table. (See Traffic Handling on Lightweight 4over6 Interfaces.)

Multiple Tunnel-Endpoint Support

Each entry can be configured with its own tunnel-endpoint address. A maximum of 32 tunnel-endpoint addresses are supported per binding table. Using the CLI, a Lightweight 4over6 binding table entry may consist of: the IPv6 tunnel address of the

CPE; the IPv4 NAT address, assigned to the client by the CPE; the ports corresponding to the IPv4 NAT address; and the IPv6 tunnel-endpoint address.

NOTE:

- Removing or deleting an active Lightweight 4over6 binding table removes all the configured tunnel-endpoint addresses as well.
- The tunnel endpoint address is mandatory for every entry.

Syntax Rules for a Binding Table

Lightweight 4over6 binding tables use the following syntax. The syntax is the same for binding tables configured on the ACOS device and for imported binding tables.

```
ipv6-tunnel-addr [ipv4-nat-addr port portnum [to portnum]] tunnel-endpoint-address
```

In each entry:

- The *ipv6-tunnel-addr* is the IPv6 address of client CPE. This is the IPv6 address of the remote end of the tunnel.
- The *ipv4-nat-addr* is the IPv4 NAT address the CPE assigns to the client. This must be a host address, not a subnet address.
- The **port** *portnum* [**to** *portnum*] is the protocol port number or range the CPE may use the as the source port in the IPv4 NAT address assigned to the client by the CPE.
- The *tunnel-endpoint-address* is the LW-4over6 IPIP Tunnel Endpoint Address.

For imported binding-table files, you also can add comments.

NOTE: You must add a semicolon (;) in front of the comment text.

After creating the binding-table file, import it onto the ACOS device. For more information about importing the file to the ACOS device, see Configuring Lw4o6

Example Binding-table File

```
; LW-4over6 Binding Table - lw-test
; LW-4over6 Number of Entries - 2
```

```
; cgnv6 lw-4o6 binding-table lw-test
3ff7::85 10.1.1.2 port 1 to 1000 3ff7::2
3ff7::85 10.1.1.2 port 20001 to 30000 2222::1
```

This example contains 2 entries. The first entry matches on CPE address `3ff7::85`, NAT address `10.1.1.2` with protocol ports in the range 1-1000, and tunnel-endpoint address `3ff7::2`. The second entry matches on CPE address `3ff7::85`, with NAT address `10.1.1.2` with port range 20001-30000, and tunnel-endpoint address `2222::1`.

Impact of Binding Table Changes on Active Traffic

If you make changes to the active binding table, the changes are applied to active traffic in 1 minute. If any Lightweight 4over6 traffic no longer matches a binding-table entry, the ACOS device handles the traffic as described above.

NOTE: The maximum amount of time required for changes to take effect is 1minute. The changes may take effect more quickly.

Traffic Handling on Lightweight 4over6 Interfaces

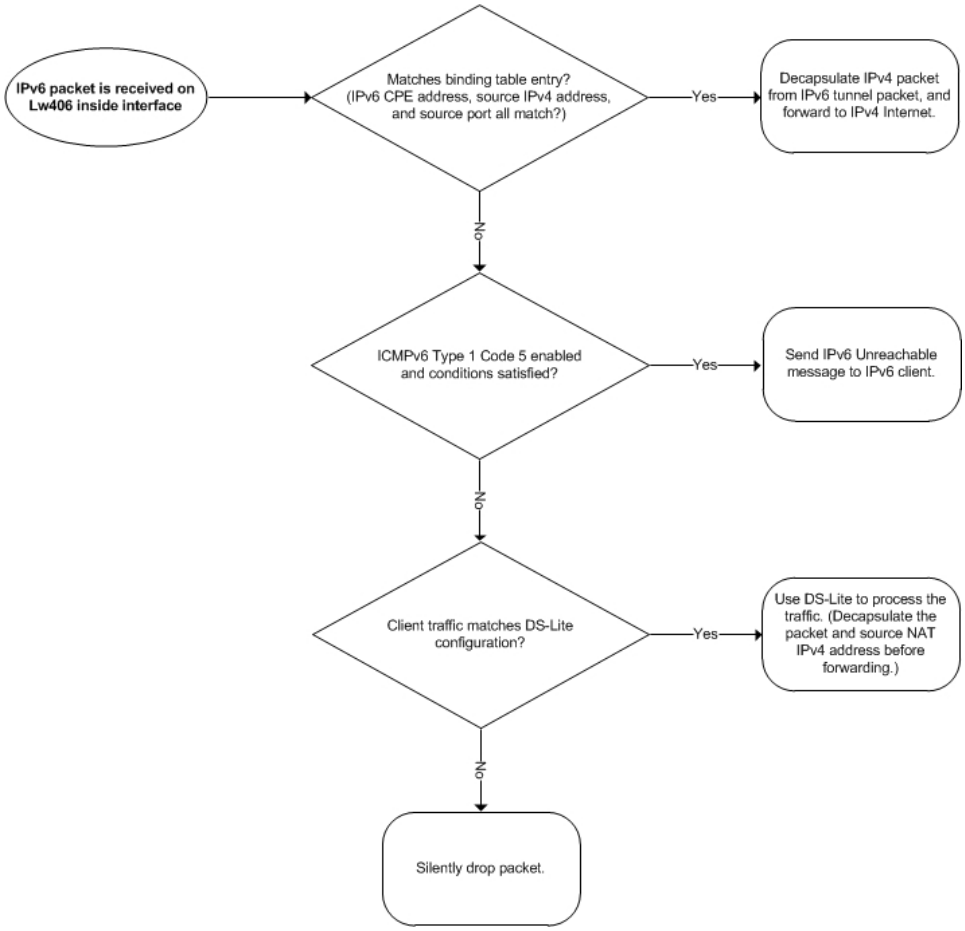
This section describes how the ACOS device processes traffic received on the interfaces you configure as the inside and outside Lightweight 4over6 interfaces.

Each subsection describes processing for inbound traffic. For example, the inside interface section describes handling of traffic that the ACOS device receives on its inside Lightweight 4over6 interface. Likewise, the outside interface section describes handling of traffic that the ACOS device receives on its outside Lightweight 4over6 interface.

Inside Lightweight 4over6 Interface

When the ACOS device receives an inbound packet on the inside Lightweight 4over6 interface, the ACOS device handles the traffic as shown in Figure 31.

Figure 31 : Packet Handing - inside Lightweight 4over6 interface

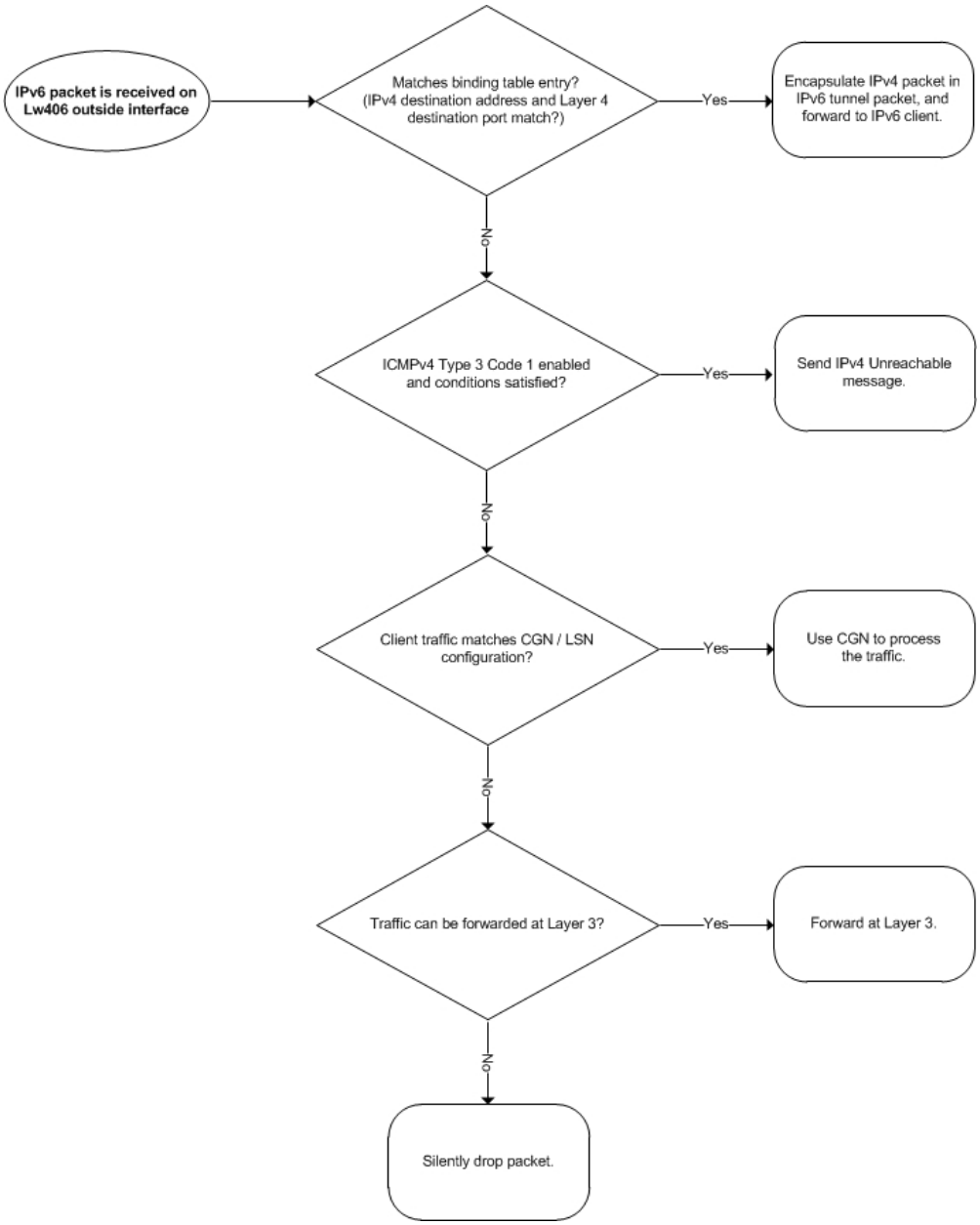


Support for ICMPv6 destination unreachable messages on the inside Lightweight 4over6 interface is optional. When enabled, messages are sent only in certain cases. For more information, see Enabling Destination Unreachable Messages for Non-matching Traffic.

Outside Lightweight 4over6 Interface

When the ACOS device receives an inbound packet on the outside Lightweight 4over6 interface, the ACOS device handles the traffic as shown in Figure 32.

Figure 32 : Packet Handing - outside Lightweight 4over6 interface



Support for ICMPv4 destination unreachable messages on the outside Lightweight 4over6 interface is optional. When enabled, messages are sent only in certain cases. For more information, see Enabling Destination Unreachable Messages for Non-matching Traffic.

Fragmentation

The following summarizes the default behavior of how ACOS handles fragmented packets for inbound and outbound traffic:

- For inbound traffic, by default, ACOS fragments oversized packet at the IPv6 tunnel level if DF bit is not set. If DF bit is set, ACOS sends an ICMP Type-3, Code 4 (Fragmentation Needed and DF Set) message.
- For outbound traffic, by default, ACOS fragments oversized IPv4 packets at the IPv4 layer if DF bit is not set. If the DF bit is set, ACOS sends an ICMP Type-3 Code 4 (Fragmentation Needed and DF Set) message.
- To change the default behavior of fragmented packets, use the **`cgnv6 lw-4o6 fragmentation`** command. For details on the complete usage of this command, see the *Command Line Reference*.

NOTE: Packet virtual reassembly is required for Carrier Grade NAT (CGN) devices to perform NAT and handle ALG traffic.

Lw4o6 access-list for Inside IPv4 Clients

Access Control List (ACL) regulates traffic going through a Lightweight 4over6 tunnel. An ACL can be applied to Lightweight 4over6 traffic from the inside client. Both an IPv4 standard ACL and an IPv4 extended ACL can be applied to Lightweight 4over6 traffic. The behavior of the ACL filtering remains the same.

If logging is enabled for ACLs, then the log will only have the IPv4 information of the Lightweight 4over6 traffic because there is no IPv4-in-IPv6 ACL.

If an ACL is configured for Lightweight 4over6, then every packet needs to be matched to the ACL. For enhanced performance, ACOS may create transparent sessions for the Lightweight 4over6 traffic. If the ACL permit rule is configured after an ACL session is created, the ACL session will be deleted upon receiving forward or reverse traffic. The transparent Lightweight 4over6 session is recreated after the ACOS device receives another Lightweight 4over6 inside packet. This simplifies the ACL filtering and improves performance so that the configured action can be taken immediately when a packet matching an existing session comes in. These sessions are visible in the CLI using the **`show session`** command.

Lw4o6 for the Port-less Protocols, like GRE

ACOS Lightweight 4over6 technology supports TCP, UDP, ICMP, and port-less protocols. In order to allow traffic from port-less protocols to use Lightweight 4over6 tunnels, an entire NAT IP address must be allocated to a single user in the Lightweight 4over6 binding table. To do this, a single binding table entry needs to be configured with the full port range from port 1 up to port 65535. For entries which do not contain the full port range, only the original ACOS Lightweight 4over6 protocols (TCP, UDP, and ICMP) are supported.

Configuring Lw4o6

Configure or Import a Lw4o6 Binding Table on the ACOS device

The Lightweight 4over6 feature uses a binding table, as described in Binding Table. For the ACOS device to be able to access the binding table, the table must be present on the ACOS device as a file.

You can add a binding-list file to the ACOS device in either of the following ways:

- Configure the file on another device (for example, on a laptop PC), then import it the file onto the ACOS device. (See Syntax Rules for a Binding Table.)
- Configure the binding list entries directly on the ACOS device, then save the configuration to save the entries into a file.

Import or Configure a Lightweight 4over6 Binding Table Using the GUI

1. Navigate to **CGN > LW-4over6 > Binding Tables**.
2. Click **Create**.
3. Enter the **Name** of the binding table.
4. Click **Add**, and enter the parameters for the Entry.
5. Click **Create**.

Import or Configure a Lightweight 4over6 Binding Table Using the CLI

You can use the commands in this section to import and create the binding table.

Configuring a Lw4o6 Binding Table

The following commands configure a binding table, and save it to a file. (For simplicity, this example shows a single entry. Binding tables can contain more than a single entry.)

```
ACOS(config)# cgnv6 lw-4o6 binding-table fw4o6-table
ACOS(config-lw-4o6)# tunnel-address 2001:10:10::9
ACOS(config-lw-4o6-ipv6)# nat-address 206.190.35.47
ACOS(config-lw-4o6-ipv6-nat)# port 1 to 30000 tunnel-endpoint-address
2001:10:10::8
ACOS(config-lw-4o6-ipv6-nat)# exit
ACOS(config-lw-4o6-ipv6)# exit
ACOS(config-lw-4o6)# tunnel-address 2001:10:10::10
ACOS(config-lw-4o6-ipv6)# nat-address 206.190.35.47
ACOS(config-lw-4o6-ipv6-nat)# port 30001 to 65535 tunnel-endpoint-address
2001:10:10::11
ACOS(config-lw-4o6-ipv6-nat)# exit
ACOS(config-lw-4o6-ipv6)# exit
ACOS(config-lw-4o6)# exit
ACOS(config)# write memory
Building configuration...
Write configuration to default startup-config
[OK]
```

Importing a Lightweight 4over6 Binding Table

Enter the following command to import a binding table:

```
ACOS(config)# import lw-4o6 table1 ftp://user:single@192.168.219.234/a.txt
```

Exporting a Lightweight 4over6 Binding Table

Enter the following command to export a binding table:

```
ACOS(config)# export lw-4o6 table1 ftp://user:single@192.168.219.234/a.txt
```

Deleting a Binding Table

The ACOS device can contain up to 10 binding tables.

Enter the following command to delete a binding table:

```
ACOS(config)# no cgnv6 lw-4o6 binding-table table1
```

The command is entered at the global configuration level of the CLI.

Activate the Binding Table

To place the binding table into effect, you must activate it.

The following command activates the binding table:

```
ACOS(config)# cgnv6 lw-4o6 use-binding-table fw4o6-table
```

NOTE: The binding-table file must be on the ACOS device. If you configured the entries directly on the ACOS device, make sure to save the configuration. Saving the configuration creates the file.

Enabling Inside Lightweight 4over6 on the Interface Connected to Clients

Enter the following command to enable the interface that is connected to clients:

```
ACOS(config)# interface ethernet 3
ACOS(config-if:ethernet:3)# lw-4o6 inside
```

Enable Outside Lw4o6 Support on the Interface Connected to the Internet

Enter the following command to enable the outside interface that is connected to the IPv4 Internet:

```
ACOS(config)# interface ethernet 3
ACOS(config-if:ethernet:3)# lw-4o6 outside
```

Additional Configuration Options

This section describes the CLI configurations.

The following topics are covered:

Configuring Additional Options Using the GUI

To configure these options using the GUI, select **CGN > LW-4over6 > LW-4over6 Global**. To configure them using the CLI, use the syntax shown in these sections or in the *CLI Reference*.

Validating Lw4o6 Binding Tables

Binding table validation checks an imported binding table and logs all the error entries into a file. If any error entries are found, a warning message indicates that errors are present in the validated binding table.

```
ACOS(config)# cgnv6 lw-4o6 binding-table-validate file-name
```

To show the error files resulting from the **lw-4o6 binding-table-validate** command, enter the following command:

```
ACOS(config)# show cgnv6 lw-4o6 binding-table-validation-log files
```

NOTE:

- The maximum number of log files that can be present at any time is 100.
- For Thunder 14045 devices, the output is displayed only for Master.
- For Thunder 7650 devices, the output is displayed only for one instance of the processing unit.

Configuring Multiple Tunnel-Endpoint Addresses for Lightweight 4over6

To configure multiple tunnel-endpoint addresses in a Lightweight 4over6 binding table, do the following:

To configure a Lightweight 4over6 binding table entry, enter the following command at the Lightweight 4over6 binding table configuration level:

```
tunnel-IPv6-address [NAT-ipv4-address port num to num ipv6-tunnel-endpoint-address]
```

NOTE: This command is entered at the configuration level. A maximum of 32 tunnel endpoint addresses can be configured. For each entry, the tunnel endpoint address is mandatory.

```
ACOS(config)# cgnv6 lw-4o6 binding-table TEST
ACOS(config-lw-4o6)# tunnel-address 3::3
ACOS(config-lw-4o6-ipv6)# nat-address 1.1.1.1
ACOS(config-lw-4o6-ipv6-nat)# port 1 to 65535 tunnel-endpoint-address 4::4
```

Configuring Access Control Lists for Lightweight 4over6 Inside Clients

To apply an ACL to Lightweight 4over6 traffic, enter the following command at the global configuration level:

```
ACOS(config)# cgnv6 lw-4o6 inside-src-access-list acl-ID
```

The *acl-num* option specifies the ACL number for the ACL to be applied to Lightweight 4over6 traffic.

CLI Configuration Example

The following example configures a Lightweight 4over6 binding table, as well as the inside and outside interfaces. It also configures an ACL for UDP traffic, and then applies the ACL to Lightweight 4over6 traffic.

The following commands configure a Lightweight 4over6 binding table named **lw4o6acl** and set it as the active Lightweight 4over6 table:

```
ACOS(config)# cgnv6 lw-4o6 binding-table lw4o6acl
ACOS(config-lw-4o6)# tunnel-address 3::3
ACOS(config-lw-4o6-ipv6)# nat-address 1.1.1.1
ACOS(config-lw-4o6-ipv6-nat)# port 1 to 65535 tunnel-endpoint-address 4::4
ACOS(config-lw-4o6-ipv6)# exit
ACOS(config-lw-4o6)# exit
ACOS(config)# cgnv6 lw-4o6 use-binding-table lw4o6acl
```

The following commands configure the inside and outside interface for Lightweight 4over6:

```
ACOS(config)# interface ethernet 1
ACOS(config-if:ethernet1)# ipv6 address 2001::aa/32
ACOS(config-if:ethernet1)# lw-4o6 inside
ACOS(config-if:ethernet1)# exit
ACOS(config)# interface ethernet 2
ACOS(config-if:ethernet2)# ip address 192.0.2.10 /12
ACOS(config-if:ethernet2)# lw-4o6 outside
ACOS(config-if:ethernet2)# exit
```

The following command configures an extended ACL to permit UDP traffic for any source on a port greater than 128 and for any destination:

```
ACOS(config)# access-list 128 permit udp any gt 128 any
```

The following command applies the IPv4 ACL number 128 to Lightweight 4over6 clients:

```
ACOS(config)# cgnv6 lw-4o6 inside-src-access-list 128
```

In this example, any Lightweight 4over6 traffic with an IPv4 UDP packet with a port number greater than 128 will be permitted, and all other traffic will be dropped.

Configuring Fragmentation Options

To change Lw4o6 fragmentation settings, use the commands described in this section.

Enabling or Disabling Fragmentation Support

Enter the following commands to change the default action performed on the packets that require fragmentation and Don’t Fragment Bit is not set:

```
ACOS(config)# cgnv6 lw-4o6 fragmentation inbound ipv6
ACOS(config)# cgnv6 lw-4o6 fragmentation outbound ipv4
```

The **inbound** option applies to inbound traffic directing packets from the "outside" interface towards the "inside" interface. The **outbound** option applies to outbound traffic directing packets from the "inside" interface to the "outside" interface.

Overriding the Don't Fragment Bit

Enter the following command to configure the Lw4o6 action to perform on packets that require fragmentation and Don't Fragment Bit is set:

```
ACOS(config)# cgnv6 lw-4o6 fragmentation inbound df-set ipv4
ACOS(config)# cgnv6 lw-4o6 fragmentation outbound df-set ipv4
```

The **ipv4** option overrides the DF bit and uses IPv4 fragmentation for oversize packets. Similarly, the **ipv6** option overrides the DF bit and uses IPv6 fragmentation for oversize packets.

Changing the Fragment Timeout

By default, Lw4o6 allows up to 60000 milliseconds (ms) between receipt of each fragment of a fragmented packet.

Enter the following command to change the fragment timeout:

```
ACOS(config)# ip  frag timeout  1000
ACOS(config)# ipv6  frag timeout  1000
```

Changing the Fragment Session Capacity

By default, Lw4o6 can queue up to 100,000 Lw4o6 packet fragments.

Enter the following command to change the queue size:

```
ACOS(config)# ip frag max-reassembly-sessions 4000
```

You can specify the maximum number of simultaneous fragmentation sessions the ACOS device will allow. The specified maximum applies to both IPv4 and IPv6.

Configuring Lightweight 4over6 for Port-less Protocols

No additional configuration changes are needed to configure Lightweight 4over6 support for port-less protocols, outside of assigning a full NAT IP address to a single user within the binding table.

CLI Configuration Example

The following example configures a Lightweight 4over6 binding table with one entry. The lone entry contains the full port range, thus allowing traffic from port-less protocols.

```
ACOS(config)# cgnv6 lw-4o6 binding-table portless
ACOS(config-lw-4o6)# tunnel-address 3::3
ACOS(config-lw-4o6-ipv6)# nat-address 1.1.1.1
ACOS(config-lw-4o6-ipv6-nat)# port 1 to 65535 tunnel-endpoint-address 4::4
ACOS(config)# cgnv6 lw-4o6 use-binding-table portless
```

Configuring Hairpin Filtering

By default, Lightweight 4over6 does not perform filtering to prevent self hairpinning. Self hairpinning occurs if traffic initiated by an inside client is routed back to itself.

You can set hairpin filtering to one of the following levels of granularity:

- **filter-all** disables all hairpinning.
- **filter-none** allows all hairpinning. This is the default.
- **filter-self-ip** blocks hairpinning to the same IP.
- **filter-self-ip-port** blocks hairpinning to the same IP and port combination.

Enter the following command to disable all hairpin filtering:

```
ACOS(config)# cgnv6 lw-4o6 hairpinning  filter-all
```

Enabling Destination Unreachable Messages for Non-matching Traffic

By default, the ACOS device does not send Destination Unreachable messages to Lightweight 4over6 clients or to servers for non-matching traffic. You can also enable ICMPv6 or ICMP Destination Unreachable messages.

Both options are disabled by default but can be enabled independently.

ICMPv6 Destination Unreachable Messages

When this option is enabled, the ACOS device can send an ICMPv6 Destination Unreachable message (type 1, code 5) to the client CPE. This type of message is sent in the following cases:

- IPv6 tunnel address matches a binding-table entry, but the source IPv4 address and source protocol port do not match
- Source IPv4 address matches a binding table entry, but the protocol port number does not match that entry
- Source IPv4 address and protocol port number match a binding table entry, but do not match the IPv6 tunnel address of that entry

When this option is enabled, it applies only to outbound IPv6 traffic received on the Lightweight 4over6 inside interface. This is CPE-to-IPv4 server traffic. The inside interface is connected to clients.

Enter the following command to enable ICMPv6 destination unreachable messages for Lightweight 4over6:

```
ACOS(config)# cgnv6 lw-4o6 no-forward-match send-icmpv6
```

This command is entered at the global configuration level of the CLI.

IPv4 ICMP Destination Unreachable Messages

When this option is enabled, the ACOS device can send an IPv4 ICMP Unreachable message, in the following cases:

- If an inbound IPv4 packet's destination IPv4 address matches a binding-table entry but not the entry's protocol port(s), the ACOS device sends an ICMP message to the IPv4 packet's sender.
- If there is no binding-table match and the packet is not otherwise filtered out (for example, by an ACL on the inbound interface), the packet is forwarded at Layer 3.

When this option is enabled, it applies only to inbound IPv4 traffic received on the Lightweight 4over6 outside interface. The outside interface is connected to the IPv4 Internet.

Enter the following command to enable ICMPv4 destination unreachable messages for Lightweight 4over6:

```
ACOS(config)# cgnv6 lw-4o6 no-reverse-match send-icmp
```

This command is entered at the global configuration level of the CLI.

Configuring the Handling of Inbound IPv4 ICMP Traffic

By default, the ACOS device handles inbound IPv4 ICMP traffic for Lightweight 4over6 sessions. Optionally, you can disable handling of this ICMP traffic.

Enter the following command to drop inbound IPv4 ICMP traffic for Lightweight 4over6 sessions:

```
ACOS(config)# cgnv6 lw-4o6 icmp-inbound drop
```

Configuring Route Redistribution

For information about configuring route redistribution, see Route Redistribution for Lightweight 4over6.

Displaying and Clearing Lw4o6 Information

Displaying Lw4o6 Information

The following commands verify the presence of the binding-table file, its active state, and its contents.

NOTE:

- To view the configured binding table, make sure that the binding table has been activated.
- For Thunder 14045 ACOS devices, the output is displayed only for master.
- For Thunder 7650 ACOS devices, the output is displayed only for one processing unit.

```
ACOS# show cgnv6 lw-4o6 binding-table files
Name                  Active   Modified
---------------------------------------
fw4o6-table           yes      no
Total: 1

ACOS# show cgnv6 lw-4o6 binding-table
Tunnel IPv6 Address                       Public Address   Start Port   End
Port
------------------------------------------------------------------------------
----
2001:10:10::9                             206.190.35.47    1
30000
2001:10:10::10                            206.190.35.47    30001
65535
```

The following command shows all the binding table log files:

```
ACOS# show cgnv6 lw-4o6 binding-table-validation-log files
```

It can also be used with the standard output modifiers:

```
ACOS# show cgnv6 lw-4o6 binding-table-validation-log files | ?
  begin    Begin with the line that matches
  include  Include lines that match
  exclude  Exclude lines that match
  section  Filter a section of output
```

The following command displays statistics for the binding table:

```
ACOS# show cgnv6 lw-4o6 binding-table statistics
```

```
LW-4over6 Binding Table Name: fw4o6-table
Tunnel IPv6 Address                        Public Address    Start Port    End
Port      Fwd Counter   Rev Counter
-------------------------------------------------------------------------------
----------------
2001:10:10::9                              206.190.35.47     1
30000         256          1272
2001:10:10::10                             206.190.35.47     30001
65535         491          33432
```

The following command displays general Lightweight 4over6 statistics:

```
ACOS(config)# show cgnv6 lw-4o6 statistics
LW-4over6 Statistics:
----------------------------
Total Entries Configured                          3
Self-Hairpinning Drops                            0
All Hairpinning Drops                             0
No-Forward-Match ICMPv6 Sent                      0
No-Reverse-Match ICMP Sent                        0
Inbound ICMP Drops                                0
Forward Route Lookup Failed                       0
Reverse Route Lookup Failed                       0
LW-4over6 Interfaces not Configured Drops         0
No Forward Binding Table Entry Match Drops        0
No Reverse Binding Table Entry Match Drops        0
ACOS(config)#
```

Table 18 describes the fields in this command's output.

Table 18 : show cgnv6 lw-4o6 statistics fields

Field	Description
Total Entries Configured	Total number of entries in the currently active binding table.
Self-Hairpinning Drops	Number of packets dropped because both the source and destination address information matched. • Both the source and destination IP addresses are the same, and match the IPv4 NAT address of any binding-table entry. For example: source IP address

Table 18 : show cgnv6 lw-4o6 statistics fields

Field	Description
	10.10.10.100:x to destination IP address 10.10.10.100:y. • Both the source and destination IP addresses are the same and match a binding-table entry, and the packet's source and destination protocol ports also match the protocol port(s) of the same bridging-table entry. For example: source IP address 10.10.10.100:x to destination IP address 10.10.10.100:x.
All Hairpinning Drops	Number of packets dropped because both the source and destination IPv4 addresses matched entries in the binding table. This counter is incremented in any of the following cases: • The source IP address matches the IPv4 NAT address of any binding-table entry. • The destination IP address matches the IPv4 NAT address of any binding-table entry.
No-Forward-Match ICMPv6 Sent	Number of times an ICMPv6 Destination Unreachable message was sent to a client CPE, because traffic from the client partially matched a binding-table entry but did not completely match any of the entries. For example, this counter is incremented if the ACOS device receives a packet whose IPv6 tunnel address does not match any binding-table entries.
No-Reverse-Match ICMP Sent	Number of times an IPv4 ICMP Destination Unreachable message was sent to an IPv4 server, because traffic from the server partially matched a binding-table entry but did not completely match any of the entries.
Inbound ICMP Drops	Number of inbound IPv4 ICMP packets that were dropped.

Table 18 : show cgnv6 lw-4o6 statistics fields

Field	Description
Forward Route Lookup Failed	Number of times client-to-server traffic was dropped because no route was available for forwarding it to the destination server.
Reverse Route Lookup Failed	Number of times server-to-client traffic was dropped because no route was available for forwarding it to the destination Lightweight 4over6 client.
LW-4over6 Interfaces not Configured Drops	Number of packets dropped due to LW-4over6 interfaces not being configured.
No Forward Binding Table Entry Match Drops	Number of packets dropped because no matching forward binding table entry was available.
No Reverse Binding Table Entry Match Drops	Number of packets dropped because no matching reverse binding table entry was available.

Displaying Lw4o6 Binding Table in the Order Configured

The parameter "**entries**" is now added to the **show cgnv6 lw-4o6 binding-table** command to show the binding table entries in the order that they are added either manually or from a file.

Clearing Lw4o6 Information

Enter the following commands to clear Lightweight 4over6 information:

```
ACOS# clear cgnv6 lw-4o6 binding-table statistics
ACOS# clear cgnv6 lw-4o6 statistics
```

The first command clears the counters in the **show cgnv6 lw-406 binding-table statistics** output. The second command clears the counters in the **show cgnv6 lw-4o6 statistics** output.

Deleting or Exporting a Binding Table Log File

The following command deletes a binding table log file, enter the following command:

```
ACOS(config)# delete cgnv6 lw-4o6-binding-table-validation-log 11
```

The following command exports a binding table log file, enter the following command:

```
ACOS(config)# export lw-4o6-binding-table-validation-log 11
ftp://user:single@192.168.219.234/a.txt
```

Route Redistribution for Lightweight 4over6

This chapter describes how to enable route redistribution for Lightweight 4over6.

The following topics are covered:

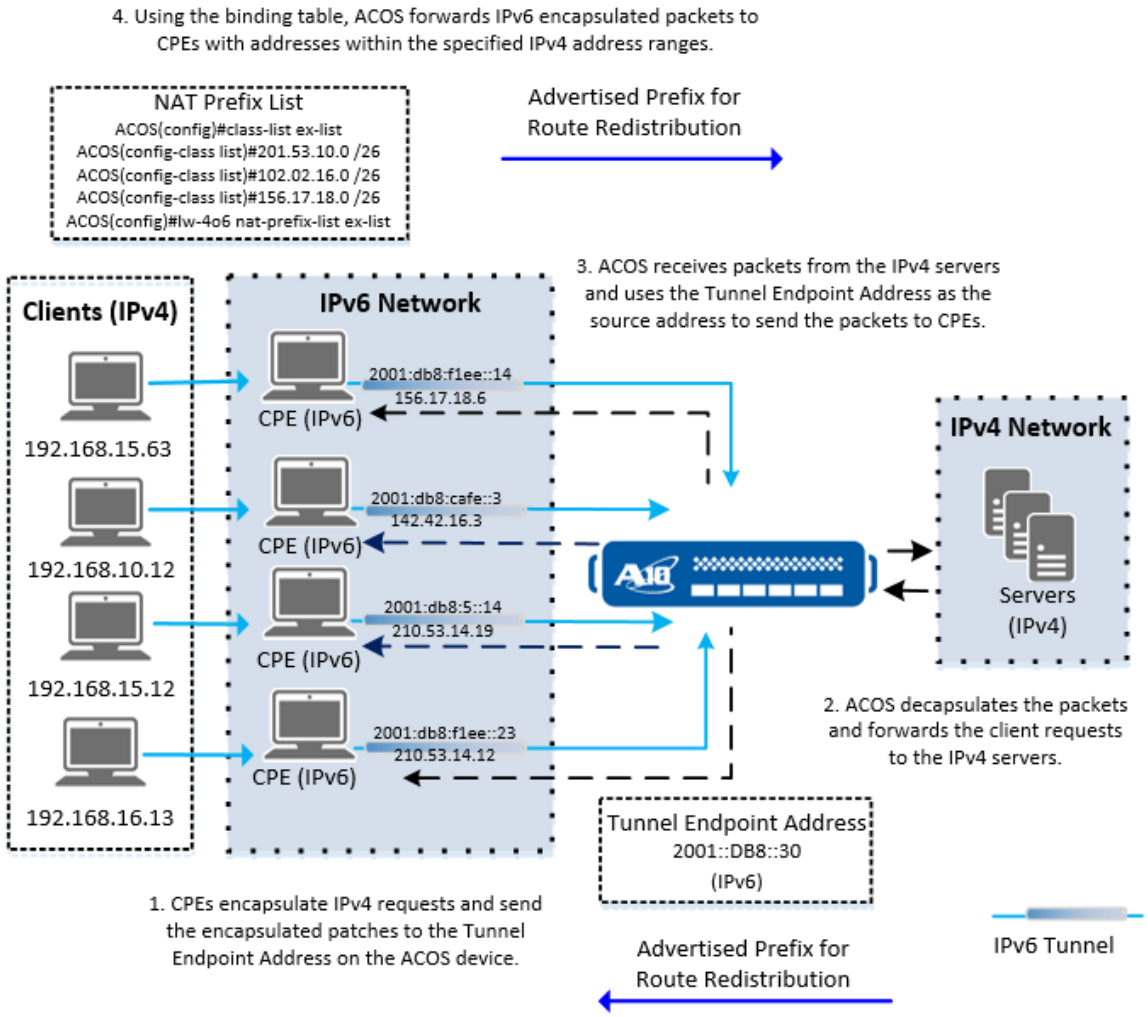
Overview

You can specify a Tunnel Endpoint Address on the ACOS device, which enables ACOS to immediately recognize all traffic from customer premise equipment (CPE) to the Tunnel Endpoint Address as Lightweight 4over6 traffic. In addition, you can configure a NAT Prefix List for Lightweight 4over6 route redistribution from the ACOS device back to multiple NAT IPv4 addresses.

Deployment Example

Figure 33 shows an example of Lightweight 4over6 route redistribution.

Figure 33 : Lightweight 4over6 – Route Redistribution



Lightweight 4over6 Route Redistribution Options

The Lightweight 4over6 configuration options include the ability to redistribute routes, based on the tunnel endpoint address or prefix range of NAT IPv4 addresses.

To prevent the ACOS device from attempting to use down links for Lightweight 4over6, you can enable a health check to periodically monitor the gateway. If ACOS detects that the gateway is down, ACOS stops sending packets to the gateway and drops Lightweight 4over6 traffic. The ACOS device continues to monitor the down

gateway and establishes the gateway for Lightweight 4over6 again when the gateway passes the health check.

The following section describes the new Lightweight 4over6 options for route redistribution. For information about general Lightweight 4over6 configurations, see Lightweight 4over6.

Tunnel Endpoint Address

You can specify a unique IPv6 address as the Tunnel Endpoint Address as part of the binding table configuration. When you configure this address, the ACOS device recognizes all traffic that is destined to the specified tunnel endpoint as Lightweight 4over6 traffic. The Tunnel Endpoint Address is used as the source address for all reverse IPv4 traffic that needs to be encapsulated in an IPv6 tunnel and sent back to the CPE.

By allowing you to specify a distinct Tunnel Endpoint Address, with the Tunnel Endpoint as the destination address, you can easily capture traffic from CPE and ensure that the ACOS device recognizes this traffic for Lightweight 4over6.

Remember the following information:

- All Lightweight 4over6 traffic must have the tunnel endpoint address as the destination address.
- The IPv6 address that is configured as the Tunnel Endpoint Address must be unique for the entire ACOS system.

NAT Prefix List

NAT prefix list is used to redistribute routes back to the IPv4 segment and it contains the public range which is used by CPE to perform NAT translation. ACOS uses that NAT prefix list to perform route redistribution and allows returned packets to find the route back to clients. On the IPv4 side (from BR to Internet), ACOS advertises NAT address ranges for traffic from internet to reach BR. Since LW-4o6 uses one-to-one mappings, each mapping will have a specific NAT IP address. User can pool these NAT addresses used by CPE into a NAT prefix range and configure it under nat-prefix-list. This ensures that aggregated routes are distributed.

When redistributing routes, you can configure gateway health monitoring before advertising LSN NAT pool prefixes. You can configure the ACOS device to perform health checks for its nexthop gateway. If a gateway goes down, the ACOS device discontinues route redistribution and stops redistributing LSN NAT pool prefixes. When the gateway has returned to an up and running state, the ACOS device will continue LSN NAT pool prefix route redistribution.

Gateway Health Monitors

You can include a gateway health check for Lightweight 4over6 route redistribution. When gateway health monitoring is configured, ACOS periodically health checks the gateways. If a gateway fails a health check, ACOS marks the gateway as down, discontinues route redistribution, and drops all current Lightweight 4over6 traffic. The ACOS device continues to monitor the down gateway and enables Lightweight 4over6 route redistribution when all of the gateways have passed the health checks. To ensure that the ACOS device does not direct Lightweight 4over6 traffic to down links, you can enable this option.

NOTE: You can configure health monitoring for a maximum of 8 gateways.

Configuring Lightweight 4over6 Route Redistribution

Configuring Lw4o6 Route Redistribution by Using the GUI

You can configure Lightweight 4over6 Route Redistribution by using the GUI.

Configuring the Tunnel Endpoint Address

1. Navigate to **CGN > LW-4over6 > Global**.
2. Click **Create**.
3. In the Tunnel IPv6 Endpoint Address field, enter an IPv6 address.

   The Tunnel Endpoint Address must be unique for the entire ACOS system.

4. (Optional) Configure other Lightweight 4over6 options.
5. Click **Create**.

Applying a NAT Prefix List

You can create a class list and apply the class list to Lightweight 4over6:

Configuring a Class List

1. Navigate to **CGN > LSN > Class Lists**.
2. Click **Create**.
3. Enter a class-list name.
4. By **Address Type**, select IPv4 and enter the respective parameters.
5. Click **Add**.
6. Repeat and for each entry.

NOTE: You can configure a maximum of 128 entries in the NAT prefix list.

7. Click **Create**.

Adding the IP Address Prefix List

1. Navigate to **CGN > LW-4over6 > Global**.
2. In NAT Prefix List, select the name of the configured class list.
3. (Optional) Configure other Lightweight 4over6 options.
4. Click **Update**.

Configuring Lw4o6 Route Redistribution by Using the CLI

Configure General Lw4o6

1. Configure or import a Lw4o6 binding table on the ACOS device.
2. Activate the binding table.

3. Enable inside Lw406 support on the ACOS data interface connected to the IPv6 CPE of clients.
4. Enable outside Lw4o6 support on the ACOS data interface connected to the IPv4 internet.

For more information, see Configuring Lw4o6.

Create a class-list of NAT IPv4 Prefixes and Apply to Global Lw4o6 Settings

The following procedures describe how to create a class list and apply the class list to Lightweight 4over6 using the CLI.

Configuring a Class List

1. Create a class-list of NAT IPv4 address prefixes to use for route redistribution, and apply the class-list to Lightweight 4over6.

```
ACOS(config)# class-list lw-4o6-nat-prefixes
ACOS(config-class list)# 192.168.15.12
ACOS(config-class list)# 192.168.16.13
ACOS(config-class list)# exit
```

2. Enter the following command to apply a class list to Lightweight 4over6 as the NAT Prefix List:

```
ACOS(config)# cgnv6 lw-4o6 nat-prefix-list lw-4o6-nat-prefixes
```

Configure Gateway Health Monitoring

This procedure is crucial to prevent ACOS from using down links for route redistribution.

The following example helps you create a health monitor for your IPv4 and IPv6 gateway and allows you to check their health before advertising LSN NAT IP prefixes by using route redistribution:

1. Enter the following command at the global configuration level to create the health monitor:

```
ACOS(config)# health monitor hm
```

2. Enter the following command to specify the health monitor type to be used:

```
ACOS(config-health:monitor)# method icmp
```

3. Enter the following command to create the gateway and apply the health monitor to it:

```
ACOS(config)# cgnv6 server lsn-health-gw 9.9.9.234
ACOS(config-real server)# health-check hm
ACOS(config)# cgnv6 lsn health-check-gateway 9.9.9.234
ACOS(config)# cgnv6 server ipv6-gateway 2001::173
ACOS(config-real server)# health-check hm
ACOS(config)# cgnv6 lsn health-check-gateway 2001::173
```

4. Enter the following command to enforce gateway health monitoring for Lightweight 4over6.

```
ACOS(config)# cgnv6 lw-4o6 health-check-gateway 2001::173
```

NOTE: If a specified gateways fails a health check, ACOS drops the Lightweight 4over6 traffic and discontinues route redistribution. Repeat this command for each gateway.

5. Enable health monitoring for one or more gateways. ACOS will periodically check each gateway and drop Lightweight 4over6 traffic if any of the gateways are marked as down.

```
ACOS(config)# cgnv6 lw-4o6 health-check-gateway 9.9.9.173
ACOS(config)# cgnv6lw-4o6 health-check-gateway 3201::172
```

After you configure this health check, the ACOS device periodically checks the health of the gateways. If a gateway is down, the NAT pool prefix routes are withdrawn.

6. Enter the **show log** command to display the following log message:

```
Warning [ACOS]:LSN: Health Check Gateway <IPv4 address | IPv6 address>
down
```

When the configured gateways are back online, the LSN NAT pool prefix routes are redistributed. The following message is logged:

```
Info  [ACOS]:LSN: All Health Check Gateways are up
```

Configure Routing Protocols and Enable Redistribution of Lw4o6 traffic

Enter the following commands to enable route redistribution to the Lightweight 4over6 Tunnel Endpoint Address or NAT Prefix List:

1. Enter the following command to configure routing protocols for Lightweight 4over6 traffic:

```
ACOS(config)# router bgp 1
ACOS(config-bgp:1)# bgp router-id 44.44.44.44
```

2. The following commands updates the destination prefix with the Lightweight 4over6 NAT Prefix List (in this example, 15.10.10.171 /32, 12.10.10.0 /24) for the specified BGP neighbor (peer):

```
ACOS(config-bgp:1)# redistribute lw4o6
ACOS(config-bgp:1)# neighbor 215.250.60.53 remote-as 2
ACOS(config-bgp:1)# neighbor 215.250.60.53 as-origination-interval 1
ACOS(config-bgp:1)# neighbor 215.250.60.53 advertisement-interval 1
ACOS(config-bgp:1)# neighbor 3ffe:60:60::53 remote-as 2
ACOS(config-bgp:1)# neighbor 215.250.60.53 maximum-prefix 1024
```

3. The following commands update the destination prefix with the Tunnel Endpoint Address (in this example, 3201::200) for the specified BGP neighbor (peer):

```
ACOS(config-bgp:1)# address-family ipv6
ACOS(config-bgp:1-ipv6)# redistribute lw4o6
ACOS(config-bgp:1-ipv6)# neighbor 3ffe:60:60::53 activate
```

Mapping of Address and Port (MAP)

The chapter describes the Mapping of Address and Port (MAP) technology.

MAP is one of the IPv6 transition mechanisms that maps an IPv4 address, prefix, or IPv4 address and port into an IPv6 address. MAP offers the capabilities in mapping between IPv6 address and IPv4 addresses and transport layer ports. MAP technology comprises two modes: MAP Translation (MAP-T) and MAP Encapsulation (MAP-E). MAP-T is a stateless form of translating packets between IPv4 and IPv6 networks. MAP-E uses an IPv4-in-IPv6 encapsulation mechanism to transport IPv4 packets over IPv6 networks.

The following topics are covered:

Overview

MAP is a stateless form of translating and encapsulating packets between IPv4 and IPv6 networks. The MAP technique builds on the Address plus Port method of stateless NAT, where each private IP is assigned a range of ports within the NAT address. Traffic is then routed based on the NAT address and port, rather than tracking each TCP and UDP flow.

MAP extends Address plus Port capabilities to leverage MAP-T in IPv4-to-IPv6 translation (and vice-versa) and MAP-E in IPv4 to IPv4-in-IPv6 encapsulation (and vice versa), across a domain that consists of MAP CE devices and a border router.

Configuration Notes

When creating a MAP-T domain, a Default Mapping Rule (DMR) must first be configured, followed by a Basic Mapping Rule (BMR). The DMR is used to map IPv4 addresses to IPv6 addresses beyond the MAP-T domain. Similarly, when creating a MAP-E domain, a Tunnel-End-Point (TEP) address must first be configured, followed by a Basic Mapping Rule (BMR). The TEP is used as the destination address for traffic from the CPE. A single TEP can be used by multiple MAP domains.

Limitations

- MAP-T does not support inter-partition configurations.
- VRRP is not supported in MAP-T.

For each BMR, a maximum number of 1024 IPv6 IPv4 prefix rule sets is supported. Support is also available for **`share-ratio`** and **`port-start`** options for the rule-ipv4-prefix in BMR. All MAP-T domains require a configured DMR and a configured BMR. **`share-ratio`** refers to the number of subscribers/CEs a public (NAT) IP is shared with. **`port-start`** refers to the beginning of the port set range to be allocated to the subscribers for each public IP. For example, in the configuration **`rule-ipv4-prefix 192.0.8.0 /24 shared-addr share-ratio 256 port-start 1024`**, there are 256 CEs sharing the NAT address. Port start defines the size of port block chunk and the range of ports (1024-65535). In this case, the Port block chunk size is 1024. CEs are 256. Each port block chunk is divided into 1024/256 groups for PSID assignment.

Prefix-rule Port Settings

1024 prefix rules are supported per domain. The IPv4 address type and port settings are configurable at the domain level only. Domain-level ipv4-address-port settings must be configured prior to configuring individual prefix-rule. Domain-level ipv4-address-port settings cannot be modified if prefix-rules are present. Optionally, for MAP-E, each prefix-rule can be configured with its own ipv4-address-port settings. If the ipv4-address-port setting is absent in a prefix-rule, then domain-level ipv4-address-port settings are applied to that rule.

Within a MAP domain, the ACOS device sits at the edge and acts as the MAP Border Relay (BR). The ACOS device uses the configured DMR and BMR to translate between IPv4 and IPv6 packet headers, or uses the TEP and BMR to perform IPv4-in-IPv6 encapsulation mechanism to transport IPv4 packets over IPv6 networks. and routes the traffic accordingly onto the respective IPv6 or IPv4 networks. Multiple ACOS devices can be supported as MAP BRs in the same MAP domain, and all MAP BR devices within the domain share the same DMR/TEP and BMR.

The shared DMR/TEP and BMR allow for a graceful failover when multiple ACOS devices are acting as MAP BRs. If one MAP BR device processes a link or health-check-gateway failure on v4, then the v6 route is withdrawn, and vice-versa. Since all other BR devices share the same DMR/TEP and BMR, other MAP BRs will continue to advertise the IPv4 address aggregated prefix on the IPv4 network and the default route for IPv6 addresses on the IPv6 network.

When configuring a MAP domain on the ACOS device, the domain configuration is global to all data interfaces. Each partition supports a maximum of 32 MAP-T domains, and statistics will be logged per domain. ACOS provides hair-pinning support for Hub & Spoke topologies when configuring MAP. Additionally, as a part of the IPv6 Migration suite, MAP runs concurrently with all other supported technologies, such as CGN, DS-Lite, Lightweight 4over6, NAT64/DNS64, and NAT46.

NOTE: To use MAP, the Customer Premise Equipment (CPE) must support MAP Customer Edge (CE) functionality.

Per Domain MTU and MSS Clamping

NOTE: This capability is only supported in MAP-T.

Per Domain MTU

To enhance performance and eliminate fragmentation, Maximum Transmission Unit (MTU) per domain is supported to configure the maximum size of each packet being transmitted as determined by Transmission Control Protocol (TCP). A packet size larger than the MTU will be fragmented. Per domain MTU configuration is mainly used for communication between the CEs and BRs.

Per domain MTU is applied to packets in the following two cases, with the assumption that the domain MTU is configured as 2000.

1. When an IPv6 packet with the size of 1800 bytes is received on the ‘map-t inside’ interface, one of the following applies:
   - If the packet is sent out as IPv6 (Hair-pinning), then the destination domain’s MTU (2000) is applied.
   - If the packet is be sent out as IPv4 from the ‘outside’ interface, then the outbound interface’s MTU (1500) is applied.
2. For packets coming from the "outside" interface (i.e. the internet), after being translated into an IPv6 packet, the destination domain’s MTU (2000) is applied.

MSS

Maximum Segment Size (MSS) sets the maximum size of a TCP segment that can be processed in a single, un-fragmented piece. The TCP MSS specifies the maximum length, in bytes, of data a single SYN or SYN-ACK packet in a TCP connection can have. The MSS does not include the TCP or IP header. In other words, MSS is derived from the MTU subtracting the bytes accounted for the TCP and IP headers. TCP only processes packets small enough to pass without being fragmented. MSS Clamping changes and lowers the MSS value in TCP SYN, reducing the packets to a size small enough to pass.

MSS and MTU are independent to each other.

MSS Clamping Methods

You can set TCP MSS clamping to be performed using one of the following methods:

- None – ACOS does not change the MSS value.
- Fixed value – ACOS changes the MSS to the length you specify.

A fixed MSS value must be less than or equal to the domain MTU.

- Subtract – ACOS reduces the MSS if it is longer than the specified number of bytes. This option sets the MSS based on the following calculations:
  - If MSS minus S is greater than the minimum value of N, subtract S from the MSS.
  - If MSS minus S is less than or equal to the minimum value of N, set the MSS to N.

  The minimum MSS value must be less than or equal to the domain MTU.

NOTE: The size of the IPv4 header and IPv6 header is 20 bytes. Since the same MSS is set in both directions, configure the MSS value to accommodate additional increase in IP header length due to packet translation.

For steps to configure MTU and MSS, see Configuring MAP-T.

MTU and MSS Configuration Notes and Limitation

- MSS Clamping is not adjusted based on the configured MTU. It is adjusted only based on per domain MSS configuration.
- Packet hit count is not updated immediately. Both FWD and REV counters are stored in the cache and are added back to domain statistics when the counter overflows or the cache is removed.
- Limitation: Domain level MTU configuration cannot monitor interfaces that handle MAP-T traffic. Henceforth, on interfaces handling MAP-T traffic, configure the MTU value to be higher than the domain MTU for all domains.

Configuring MAP

When configuring MAP, the domain is configured first, followed by the requisite DMR (for MAP-T) and TEP (for MAP-E), and then the BMR. When configuring the BMR, there are three possible address assignment options. A CE can be assigned either an IPv4 prefix of a NAT address, a single IPv4 NAT address, or a single IPv4 NAT address that is shared with other CEs. When the MAP domain configuration is complete, MAP Translation can be enabled on an interface.

When configuring a CE address assignment, the Embedded Address (EA) bits length needs to be specified if assigning an IPv4 NAT address prefix. The **share-ratio** and **port-start** parameters must be specified when assigning a shared IPv4 NAT address.

Optionally, health checks, route redistribution, and fragmentation and ICMP error notifications can be configured for MAP. Health checks can be configured on up to 4 gateways per domain. Gateway health checks allow for a route to be withdrawn from a BR if a link failure is detected. Route redistribution is configured at the router configuration level and can be useful as the default communication between CEs goes through a BR. Fragmentation allows for oversize packets to be fragmented or dropped.

The default behavior is to fragment oversize packages. For IPv4 packets, if the “DF” flag is set, then an ICMP error message will be sent by default. An ICMP error message can be configured optionally for IPv4 packets without a “DF” flag set. For IPv6 packets, an ICMPv6 error message can be configured. Note that fragmentation configurations are not domain specific and apply to all MAP domains within a partition.

Configuring MAP-T

- To configure a MAP-T domain, enter the following command at the global configuration. This command changes the CLI to the Map-T domain configuration level.

```
ACOS(config)# cgnv6 map translation domain 11
```

NOTE: At the MAP domain configuration level, the DMR, BMR, and health check gateway can be configured.

- To create a description for the MAP-T domain, enter the following command at the MAP-T domain configuration level:

```
ACOS(config-map-t-domain)# description mdomain1
```

- To configure the DMR, enter the following command:

```
ACOS(config-map-t-domain)# default-mapping-rule
ACOS(config-map-t-domain-dmr)#
```

- To configure the IPv6 prefix used for the DMR, enter the following command at the DMR configuration level. This command configures the IPv6 prefix used for converting IPv4 addresses to IPv6.

  ```
  ACOS(config-map-t-domain-dmr)# rule-ipv6-prefix 190::1:120:2/32
  ```

  After configuring the DMR IPv6 prefix, exit the DMR configuration level to return to the MAP-T domain configuration level.

- To configure the BMR, enter the following command:

  ```
  ACOS(config-map-t-domain)# basic-mapping-rule
  ACOS(config-map-t-domain-bmr)#
  ```

- The following commands are used to configure the address assignment schemes. Respectively, the following commands are for: assigning a CE an IPv4 prefix of a NAT address; assigning a CE a single IPv4 NAT address; or assigning a CE a shared IPv4 NAT address.

  ```
  ACOS(config-map-t-domain-bmr)# rule-ipv4-address-port-settings single-addr
  ```

  The **share-ratio** parameter refers to the number of CEs/subscribers sharing a single public (NAT) IP. The **port-start** parameter refers to the starting port of the port set range which is shared among the “share-ratio” number of CPE/subscribers. Both the **share-ratio** and **port-start** parameters must be in the values of the power of 2. The **share-ratio** parameter can be up to 65,536 while the **port-start** parameter can be up to 32,768.

- To create a name for the prefix rule, enter the following command at the BMR configuration level:

  ```
  ACOS(config-map-t-domain-bmr)# prefix-rule rule1
  ACOS(config-map-t-domain-bmr-prefix-rule)#
  ```

- To specify the prefix rule for the IPv6 prefix and IPv4 prefix to be used by the CE, enter the following command:

  ```
  ACOS(config-map-t-domain-bmr)# prefix-rule rule1
  ACOS(config-map-t-domain-bmr-prefix-rule)# rule-ipv6-prefix 3ffe:1::/64 rule-ipv4-prefix 192.0.8.0 /24
  ```

Exit the BMR configuration level to return to the MAP-T domain configuration level.

- From here, a gateway health check can be configured by entering the following command:

```
ACOS(config-map-t-domain)# health-check-gateway 192.0.8.0
```

- To configure the route withdraw behavior when a gateway health check detects a link failure, enter the following command:

```
ACOS(config-map-t-domain)# health-check-gateway withdraw-route all-link-failure
```

Exiting out of the MAP-T domain configuration level completely, enter an interface configuration level to enable MAP-T. These commands are not available on the management interface.

- Enabling MAP-T on the inside is to enable MAP-T on the interface connected the CEs. Enabling MAP-T on the outside is to enable MAP-T on the interfaces connected to the IPv4 internet.

```
ACOS(config)# interface ethernet 1
ACOS(config-if:ethernet:1)# ipv6 address 3001::1/16
ACOS(config-if:ethernet:1)# map inside
ACOS(config-if:ethernet:1)# map outside
```

- Optionally, you can configure fragmentation of oversized packets. For oversized IPv6 packets, the default is to fragment the packet. The other options are to drop the packets, or send an ICMPv6 error message that the packet is too big. Oversized IPv4 packets can also be dropped silently. If the original packet has the DF flag set, an ICMP error message is sent by default, although you can still choose to drop or fragment those packets.

```
ACOS(config)# cgnv6map translation fragmentation outbound drop
ACOS(config)# cgnv6map translation fragmentation inbound drop
ACOS(config)# cgnv6map translation fragmentation inbound df-set send-icmp
```

- To configure the MTU value, use the following commands:

```
ACOS(config)# cgnv6 map translation domain test
ACOS(config-map-t-domain)# mtu 1200
```

- To configure the MSS value, use the following commands:

```
ACOS(config)# cgnv6 map translation domain test
ACOS(config-map-t-domain)# tcp mss-clamp fixed
```

Configuring MAP-E

- To configure a MAP-E domain, enter the following command at the global configuration. This command changes the CLI to the Map-E domain configuration level.

```
ACOS(config)# cgnv6 map encapsulation domain 11
```

NOTE: At the MAP domain configuration level, the TEP, BMR, and health check gateway can be configured.

- To create a description for the MAP-E domain, enter the following command at the MAP-E domain configuration level:

```
ACOS(config-map-e-domain)# description mdomain2
```

- To configure the draft format for packet construction as the draft-03 format, enter the following command:

```
ACOS(config-map-e-domain)# format draft-03
```

- To configure the tunnel endpoint address, enter the following command:

```
ACOS(config-map-e-domain)# tunnel-endpoint-address 4001:abcd::1
```

- To configure the BMR, enter the following command:

```
ACOS(config-map-e-domain)# basic-mapping-rule
ACOS(config-map-e-domain-bmr)#
```

- The following commands are used to configure the address assignment schemes. Respectively, the following commands are for assigning a CE an IPv4 prefix of NAT addresses; assigning a CE a single IPv4 NAT address; or assigning a CE a shared IPv4 NAT address.

```
ACOS(config-map-e-domain-bmr)# rule-ipv4-address-port-settings shared-
addr share-ratio 256 port-start 1024
```

The **share-ratio** parameter refers to the number of CEs/subscribers sharing a single public (NAT) IP. The **port-start** parameter refers to the starting port of the port set range which is shared among the "share-ratio" number of CPE/subscribers. Both the **share-ratio** and **port-start** parameters must be in the values of the power of 2. The **share-ratio** parameter can be up to 65,536 while the **port-start** parameter can be up to 32,768.

- To create a name for the prefix rule and to specify the prefix rule for the IPv6 prefix and IPv4 prefix to be used by the CE, enter the following commands at the BMR configuration level:

```
ACOS(config-map-e-domain-bmr)# prefix-rule rule1
ACOS(config-map-e-domain-bmr-prefix-rule)# rule-ipv6-prefix
2002:abcd::/32 rule-ipv4-prefix 8.8.8.8 /32 single-addr
ACOS(config-map-e-domain-bmr)# prefix-rule 2
ACOS(config-map-e-domain-bmr-prefix-rule)# rule-ipv6-prefix
2003:abcd::/32 rule-ipv4-prefix 8.8.8.9 /32
ACOS(config-map-e-domain-bmr)# prefix-rule 3
ACOS(config-map-e-domain-bmr-prefix-rule)# rule-ipv6-prefix
2004:abcd::/32 rule-ipv4-prefix 8.8.8.10 /32
```

Exit the BMR configuration level to return to the MAP domain configuration level.

- From here, a gateway health check can be configured by entering the following command:

```
ACOS(config-map-e-domain)# health-check-gateway 192.0.8.0
```

- To configure the route withdraw behavior when a gateway health check detects a link failure, enter the following command:

```
ACOS(config-map-e-domain)# health-check-gateway withdraw-route all-link-
failure
```

Exiting out of the MAP-E domain configuration level completely, enter an interface configuration level to enable MAP-E. These commands are not available on the management interface.

- Enabling MAP-E on the inside is to enable MAP-E on the interface connected the CEs. Enabling MAP-E on the outside is to enable MAP-E on the interfaces connected to the IPv4 internet.

```
ACOS(config)# interface ethernet 1
ACOS(config-if:ethernet:1)# ipv6 address 3001::1/16
ACOS(config-if:ethernet:1)# map inside
ACOS(config-if:ethernet:1)# map outside
```

- Optionally, you can configure fragmentation of oversized packets. For details, see the *Command Line Reference*.

```
ACOS(config)# cgnv6map encapsulation fragmentation outbound drop
ACOS(config)# cgnv6map encapsulation fragmentation inbound drop
ACOS(config)# cgnv6map encapsulation fragmentation inbound df-set send-icmp
```

Configuration Example

The following example configures a MAP-E domain with the gateway health check, and enables the MAP-E domain on the interfaces.

```
!
interface ethernet 1
ipv6 address 3001::1/16
map inside
enable
!
interface ethernet 2
ip address 192.0.20.5 /24
map outside
  enable
!
health monitor gateway
  method icmp
!
cgnv6 server gateway1 192.0.20.100
  health-check gateway
!
cgnv6 map encapsulation domain 2
  format draft-03
  tunnel-endpoint-address 4001:abcd::1
  health-check-gateway 192.0.20.100
  health-check-gateway withdraw-route all-link-failure
```

```
  basic-mapping-rule
    rule-ipv4-address-port-settings shared-addr share-ratio 256 port-start
1024
    prefix-rule 1
      rule-ipv6-prefix 2002:abcd::/32 rule-ipv4-prefix 8.8.8.8 /32 single-
addr
    prefix-rule 2
      rule-ipv6-prefix 2003:abcd::/32 rule-ipv4-prefix 8.8.8.9 /32
    prefix-rule 3
      rule-ipv6-prefix 2004:abcd::/32 rule-ipv4-prefix 8.8.8.10 /32
    prefix-rule 4
      rule-ipv6-prefix 2005:abcd::/32 rule-ipv4-prefix 8.8.9.0 /24 prefix-
addr ea-length 4
```

Displaying and Clearing MAP Information

1. Use the following show commands to view the MAP-E domain configurations, as well as MAP-E traffic statistics. They can be used to view the domain configuration or traffic statistics for a specific domain or for all domains:

```
ACOS# show cgnv6 map encapsulation domaindomain1
ACOS# show cgnv6 map encapsulation statistics domain1
```

2. To clear MAP-E traffic statistics, use the following command:

```
ACOS# clear cgnv6 map encapsulation statisticsdomain1
```

Stateless NAT46

This chapter describes stateless NAT46 and how to configure it on the ACOS device.

The following topics are covered:

Overview

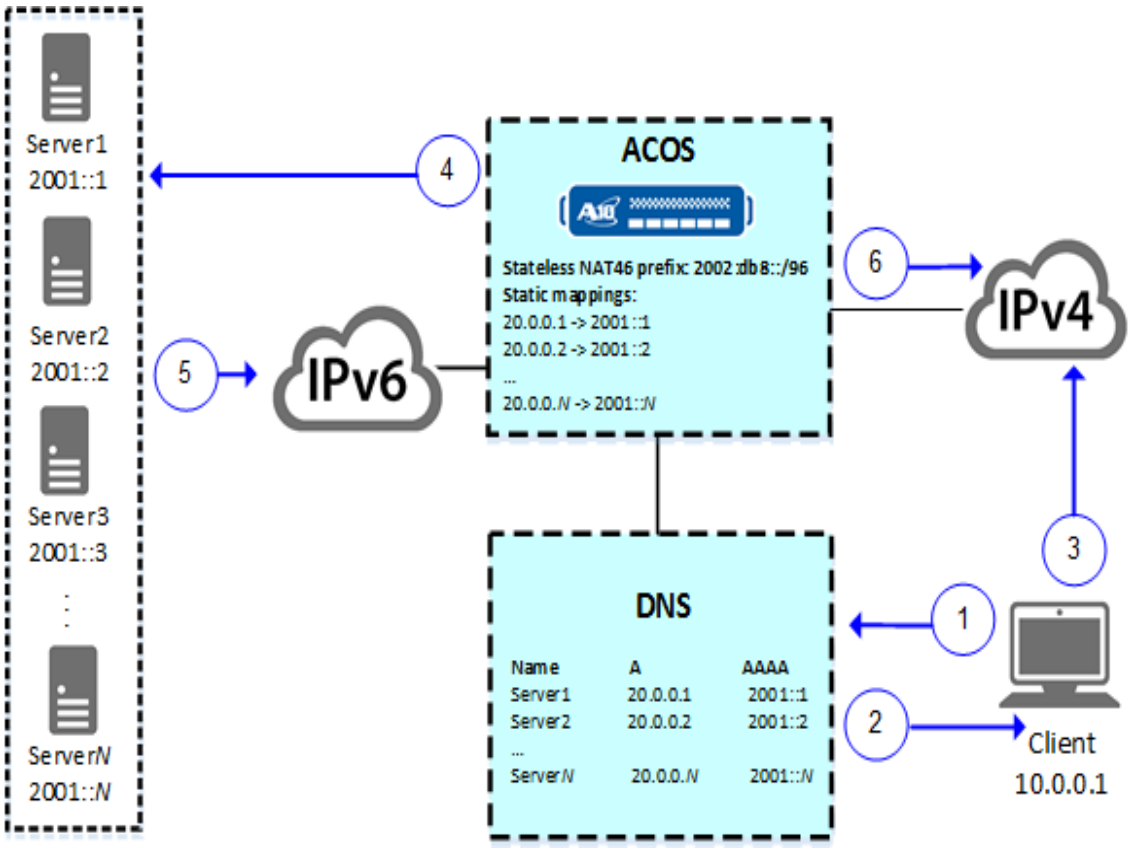
Stateless NAT46 enables IPv4 clients to reach IPv6 servers, without maintaining per-connection information on the ACOS device.

NOTE: Stateless NAT46 is based on the *NAT46 considerations, draft-liu-behave-nat46-02* RFC.

Stateless NAT46 uses statically configured IPv4-IPv6 mappings. When an IPv4 client sends a request to a server, the destination address of the request is an IPv4 address. If the destination IPv4 address is statically mapped to the server's IPv6 address, stateless NAT46 NATs the request and forwards it to the server.

Figure 34 illustrates an example of a stateless NAT46 deployment, the traffic flow for an IPv4 client's request, and the IPv6 server's response.

Figure 34 : Stateless NAT46



In this example, the client-server traffic between client 10.0.0.1 and IPv6 server 2001::4 is translated by stateless NAT46 as follows:

1. IPv4 client's browser sends a DNS request for the IPv4 address of "Server4".
2. The DNS server replies with IPv4 address 20.0.0.4.
3. The client sends an HTTP/HTTPS request to 20.0.0.4.
4. On the ACOS device, the stateless NAT46 configuration contains a static IPv4-IPv6 mapping of 20.0.0.4 to 2001::4.

   ACOS translates the client's IPv4 request into an IPv6 request with the following address translations:

   - The source address is translated to 2002:db8::a00:1 and consists of the following parts:

- The 96-bit stateless NAT46 prefix: 2002:db8::/96
- The 32-bit IPv4 address converted to hexadecimal: a00:1

- The destination IPv4 address is translated to 2001::4.

  ACOS sends an IPv6 request to 2001::4.

5. The server sends an IPv6 reply to 2002:db8::a00:1.
6. ACOS translates the server reply into IPv4 and forwards it to the client.

NOTE:

- For simplicity, the DNS server is depicted as an IPv4 and IPv6 DNS server. Separate IPv4 and IPv6 servers also can be used. The requirement for stateless NAT46 is for IPv4 clients to get the IPv4 addresses of the servers from DNS or some other mechanism.
- The IPv6 DNS is shown based on the assumption that some in the network, IPv6 clients that do not use stateless NAT46 might need to get the server's IPv6 addresses.

Stateless NAT46 Prefix

Stateless NAT46 translates an IPv4 client's address into an IPv6 address by combining the stateless NAT46 prefix that is configured on the ACOS device with the client's IPv4 address:

stateless_NAT46_prefix:client_IPv4_address

The stateless NAT46 prefix must be 96 bits long. This leaves 32 bits for the client's IPv4 address.

In this figure, the stateless NAT46 prefix is 2002:db8::/96, and the IPv4 client's address is 10.0.0.1. The ACOS device translates the client's IPv4 address into the 2002:db8::a00:1 IPv6 address.

NOTE: There is no default stateless NAT46 prefix, so you must configure it.

Mapping

Individual mappings or ranges of mappings can be configured on each partition. When configuring a range, specify the first mapping in the range and how many mappings to create. The ACOS device automatically creates the specified number of mappings. Each individual mapping in the range counts as one of the supported mappings. Depending on the size of the system memory, the number of indivdual mappings supported is as follows:

- If the system memory is smaller than 16GB, then 1024 individual mappings are supported per partition.
- If the system memory is greater then 16GB, then 8K (8* 1024) mappings are supported per partition.

The IPv4 and IPv6 addresses for each additional mapping are incremented by 1 over the previous mapping. For example, if you specify the following mapping and a quantity of 10:

- `20.0.0.1 -> 2001::1`

The ACOS device creates the following mappings:

- `20.0.0.1 -> 2001::1`
- `20.0.0.2 -> 2001::2`
- `20.0.0.3 -> 2001::3`
- `20.0.0.4 -> 2001::4`
- `20.0.0.5 -> 2001::5`
- `20.0.0.6 -> 2001::6`
- `20.0.0.7 -> 2001::7`
- `20.0.0.8 -> 2001::8`
- `20.0.0.9 -> 2001::9`
- `20.0.0.10 -> 2001::a`

NAT46 Mappings Between Shared and L3V Partitions

Inter-partition support for NAT46 mappings is available to improve scalability and performance, namely support between the shared and L3V partitions.

The following summarizes how different prefixes (intrinsic and inter-partition) are used in different partitions:

- On each partition (shared or L3V), use the **cgnv6 nat46-stateless prefix** option to define an intrinsic prefix to handle its own NAT46 traffic. To configure a prefix for prefix advertisement, use the **vrid** sub-option.
- On the shared partition:
  - Use the **cgnv6 nat46-stateless partition-prefix** command to define a prefix that handles inter-partition NAT46 traffic going to L3V partitions.
  - Use the **cgnv6 nat46-stateless static-dest-mapping** option with the **shared** sub-option to indicate if a mapping is shared with other partitions. The **shared** sub-option is only available at the shared partition.
  - A mapping defined in the shared partition configured with the **shared** sub-option takes effect for all L3V partitions, unless specified otherwise in a given L3V partition.
- On L3V partitions:
  - The inter-partition prefix defined in the shared partition is used to handle inter-partition traffic received from the shared partition. This enables prefix advertisement to upstream routers.
  - Use the **cgnv6 nat46-stateless static-dest-mapping** option with the **to-shared** sub-option to indicate any traffic matching this mapping to be sent through the shared partition. The **to-shared** sub-option is only available at L3V partitions.
  - A mapping defined in an L3V partition with the **to-shared** sub-option over-writes the mapping defined in the shared partition if the two are in conflict.

Configuration Notes

- If vrid is configured for a prefix, then only vrid-active ACOS devices will advertise this prefix. If there is VRID configuration configured for mapping, do not configure

a vrid which conflicts with the prefix vrid if both exist.

- For chassis platforms, there must be “ip nat inside” configured on the inside interface in order forNAT46 traffic to be redistributed to the master and the blade.

Configuration Example 1: NAT46 Destination Mapping in Shared Partition

NAT46 destination mappings are defined in the shared partition.

From the Shared partition:

Use the following command to configure a NAT46 prefix for L3V(p4) inter-partition NAT46 traffic:

```
ACOS(config)# cgnv6 nat46-stateless partition-prefix p4 46::/96
```

Use the following command to define the destination mapping to be shared with other partitions:

```
ACOS(config)# cgnv6 nat46-stateless static-dest-mapping 60.1.1.111
2060::108 count 1 shared
```

Configuration Example 2: NAT46 Destination Mapping in L3V Partition

NAT46 destination mappings are defined in the L3V partition.

From the Shared partition:

Use the following command to configure a NAT46 prefix for L3V(p4) inter-partition NAT46 traffic:

```
ACOS(config)# cgnv6 nat46-stateless partition-prefix p4 46::/96
```

From the L3V partition:

Use the following command to define the destination mapping L3V partition (p4) that any matching traffic will be sent through the shared partition using the configured partition prefix:

```
ACOS(config)# cgnv6 nat46-stateless static-dest-mapping 60.1.1.111
2060::108 count 1 to-shared
```

Configuration Example 3: NAT46 Destination Mapping in Both Shared and L3V Partition

NAT46 destination mappings are defined in both the shared and L3V partition. The mapping defined in the L3V partition takes precedence over the mapping defined in the shared partition.

From the Shared partition:

Use the following command to configure a NAT46 prefix for L3V(p4) inter-partition NAT46 traffic:

```
ACOS(config)# cgnv6 nat46-stateless partition-prefix p4 46::/96
```

Use the following command to define the destination mapping to be shared with other partitions:

```
ACOS(config)# cgnv6 nat46-stateless static-dest-mapping 60.1.1.111
2060::109 count 1 shared
```

From the L3V partition:

Use the following command to define the destination mapping L3V partition (p4) that any matching traffic will be sent through the shared partition using the configured partition prefix:

```
ACOS(config)# cgnv6 nat46-stateless static-dest-mapping 60.1.1.111
2060::108 count 1 to-shared
```

Configuration Example 4: Two Prefixes for One L3V Partition

There are two prefixes for an L3V partition handling inter-partition NAT46 traffic and normal NAT46 traffic simultaneously. If the NAT46 traffic is inter-partition traffic, the partition prefix defined in shared partition is used. If the NAT46 traffic is not inter-partition traffic, then the NAT46 prefix defined in L3V partition is used.

From the Shared partition:

Use the following command to configure a NAT46 prefix for L3V(p4) inter-partition NAT46 traffic:

```
ACOS(config)# cgnv6 nat46-stateless partition-prefix p4 46::/96
```

From the L3V partition:

Use the following command to configure a NAT46 prefix for its own NAT46 traffic:

```
ACOS(config)# cgnv6 nat46-stateless partition-prefix p4 46::/96
```

Use the following command to define the destination mapping L3V partition (p4) that any matching traffic will be sent through the shared partition using the configured partition prefix:

```
ACOS(config)# cgnv6 nat46-stateless static-dest-mapping 60.1.1.111
2060::108 count 1 to-shared
```

Use the following command to define the destination mapping L3V partition p4 for its NAT46 traffic:

```
ACOS(config)# cgnv6 nat46-stateless static-dest-mapping 60.1.1.112
2061::108 count 1
```

Packet Fragmentation

By default, the ACOS device uses the following fragmentation settings for stateless NAT46:

- Inbound IPv6-to-IPv4 traffic – If the ACOS device receives an oversize IPv6 packet, the device drops the packet and sends an ICMPv6 error message.
- Outbound IPv4-to-IPv6 traffic – Oversize IPv4 packets from the client are fragmented, and the fragments are encapsulated into separate IPv6 packets.
- Don’t Fragment bit – If an oversize IPv4 packet from a client has the Don't Fragment bit set, the ACOS device drops the packet and sends an ICMP error message.

These settings are configurable.

NOTE: Packet virtual reassembly is required for Carrier Grade NAT (CGN) devices to perform NAT and handle ALG traffic.

Configuring Stateless NAT46

Configuring Stateless NAT46 by Using the GUI

You can configure stateless NAT64 by using the GUI.

Configuring Static Mappings

To configure the static mappings:

1. Navigate to **CGN > Stateless NAT46 > Static Mappings**.
2. Click **Create**.
3. Enter the lowest IPv4 server address to which IPv4 clients will send requests.
4. Enter the server's IPv6 address.
5. In **Count**, specify how many mappings to create.

   The IPv4 and IPv6 addresses of each mapping are incremented by 1 over the previous mapping. For more information, see Mapping. If you do not specify a count, only 1 mapping is created.
6. (Optional) To assign the mappings to VRRP-A, select the **VRID**.
7. Click **OK**.

Configuring the Prefix and Fragmentation Settings

To configure the prefix and fragmentation settings:

1. Navigate to **CGN > Stateless NAT46 > Global**.
2. Enter the prefix that will be used as the higher-order bits of the client's IPv6 address.

   For more information, see Stateless NAT46 Prefix.
3. Enter the prefix length.
4. Complete one of the following tasks:

- To change fragmentation settings, proceed to (Optional) Modify the stateless NAT46 fragmentation settings:.
- Click **OK** to complete the configuration.

5. (Optional) Modify the stateless NAT46 fragmentation settings:
   a. In **Direction**, select the traffic direction.
   b. Select an action.
   c. To specify the ACOS response to IPv4 packets that have the Don't Fragment bit set, select DF Set and select an action.
   d. Click **Add**, and then **Update**.

Configuring Stateless NAT46 by Using the CLI

Configure IPv6 Prefix Used as Higher-order Bits of Client IPv6 Addresses

To configure the 96-bit IPv6 prefix that will be used as the higher-order bits of client IPv6 addresses, enter the following command to configure the IPv6 prefix:

```
ACOS(config)# cgnv6 nat46-stateless prefix 2002:db8::/96
```

The IPv6 prefix is used as the higher-order bits of the client's IPv6 address. For more information, see Stateless NAT46 Prefix.

Configure Static IPv4-IPv6 Mappings for IPv6 Servers Reached by IPv4 Clients

To configure static IPv4-IPv6 mappings for the IPv6 servers that will be reached by IPv4 clients, enter the following command:

```
ACOS(config)# cgnv6 nat46-stateless static-dest-mapping 1.1.1.1 2002:db8::
```

Change the Fragmentation Settings (Optional)

1. Enter the following command to change fragmentation support for inbound IPv6-to-IPv4 traffic:

   ```
   ACOS(config)# cgnv6 nat46-stateless fragmentation inbound drop
   ```

2. Enter the following command to change fragmentation support for outbound

IPv4-IPv6 traffic:

```
ACOS(config)# cgnv6 nat46-stateless fragmentation outbound drop
```

3. Enter the following command to change fragmentation support for IPv4 packets that have the Don't Fragment bit set:

```
ACOS(config)# cgnv6 nat46-stateless fragmentation outbound df-set drop
```

Additional Configuration Options

The following sections describe additional configuration options.

The following topics are covered:

Configuring TCP Maximum Segment Size Clamping

The TCP maximum segment size (MSS) specifies the maximum length, in bytes, of data that one SYN or SYN-ACK packet in a TCP connection can have. The MSS does not include the TCP or IP header.

MSS Clamping Methods

You can set TCP MSS clamping for Stateless NAT46 to be performed using one of the following methods:

- None – ACOS does not change the MSS value.
- Fixed value – ACOS changes the MSS to the specified length.
- Subtract – ACOS reduces the MSS if the value is greater than the specified number of bytes.

This option sets the MSS based on the following calculations (S - Value to subtract from the maximum MSS Clamping value and N - Minimum value of the MSS Clamping):

- If MSS minus S is greater than N, subtract S from the MSS.
- If MSS minus S is less than or equal to N, set the MSS to N.

By default, the subtract method of MSS clamping is used with the following values:

S = 40 bytes

N = 576 bytes

Using these values, the default MSS clamping calculations are as follows:

- If MSS minus 40 is greater than 576, subtract 40 from the MSS.
- If MSS minus 40 is less than or equal to 416, set the MSS to 576.

Changing the MSS Clamping Method

Enter the following command to change the MSS clamping method for Stateless NAT46 to a fixed maximum value of 120:

```
ACOS(config)# cgnv6 nat46-stateless tcp mss-clamp fixed 120
```

Displaying and Clearing Stateless NAT46 Statistics

1. Enter the following command to display stateless NAT46 statistics:

```
ACOS# show cgnv6 nat46-stateless statistics
Stateless NAT46 Statistics:
---------------------------
Outbound IPv4 packets received           10
Outbound IPv4 packets dropped            0
Outbound IPv4 fragment packets received  0
Outbound IPv6 destination unreachable    0
Outbound IPv6 packets fragmented         0
Inbound IPv6 packets received            101
Inbound IPv6 packets dropped             0
Inbound IPv6 fragment packets received   0
Inbound IPv4 destination unreachable     0
Inbound IPv4 packets fragmented          0
Packet too big                           0
Fragment process error                   0
ICMPv6 to ICMP                           1
ICMPv6 to ICMP error                     0
ICMP to ICMPv6                           0
```

```
ICMP to ICMPv6 error                    0
HA is standby                           0
```

2. Enter the following command to clear stateless NAT46 statistics:

```
ACOS# clear cgnv6 nat46-stateless statistics
```

Translating IPv6 Prefixes by Using NPTv6

This chapter provides information on how to configure NPTv6 translation with an ACOS device to manage your network traffic.

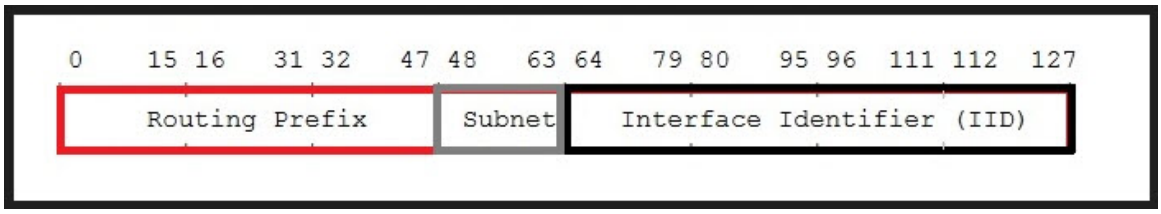
The following topics are covered:

Overview

You can configure NPTv6 translation with an ACOS device to manage your network traffic. This feature supports the requirements in RFC 6296.

Figure 35 shows the different parts of an IPv6 IP address.

Figure 35 : Enumeration of an IPv6 Address (RFC 6296)



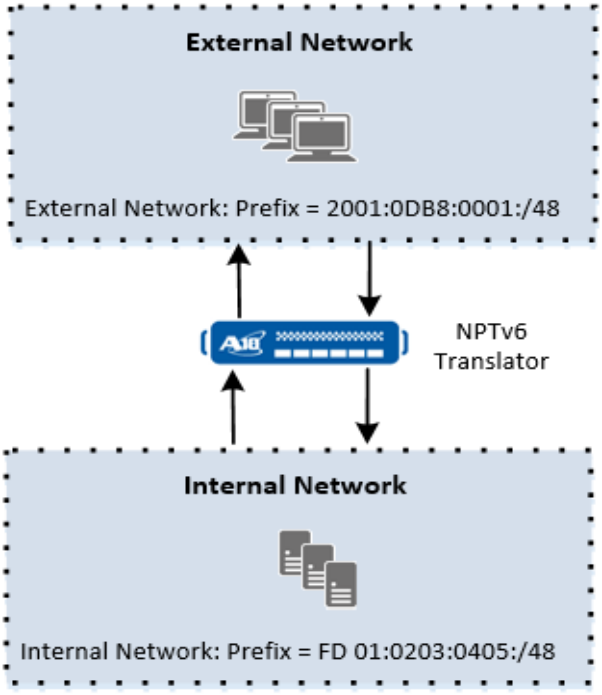
For example, you have an internal host with an IP address that has the `FD01:0203:0405:/48` prefix. When the packet moves from an internal to an external network, the source address is translated to an IP address with the `2001:0DB8:0001:/48` external prefix. When the traffic moves from an external to an internal network, the prefix changes from `2001:0DB8:0001:/48` to `FD01:0203:0405:/48`. This example is illustrated in . All the graphics in this chapter are based on the graphics in

NOTE: All the graphics in this chapter are based on the graphics in the RFC 6296.

You can configure prefix translation in one of the following ways:

- Between an internal and an external network.

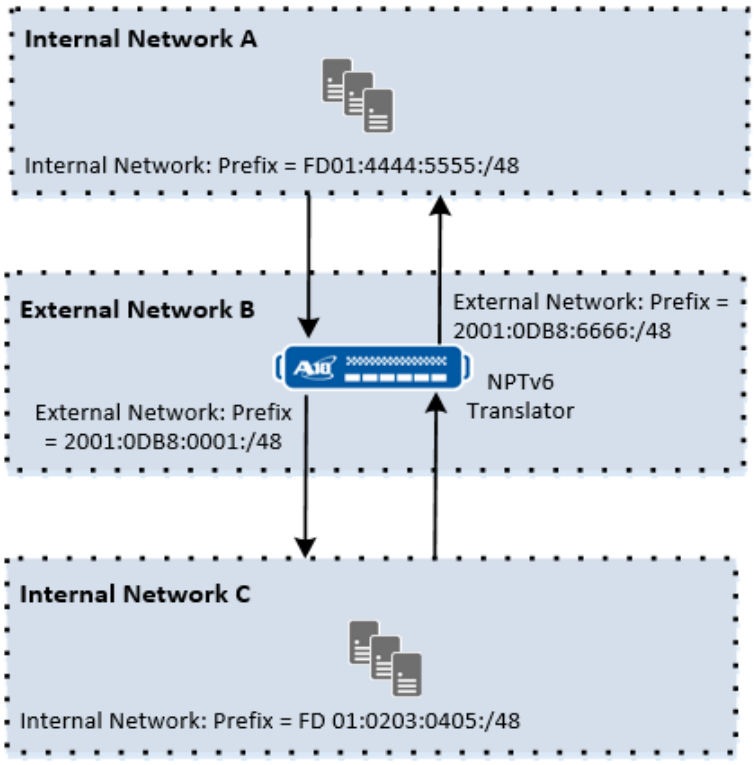
Figure 36 : Prefix Translation Between an Internal and an External Network



- Between two private networks

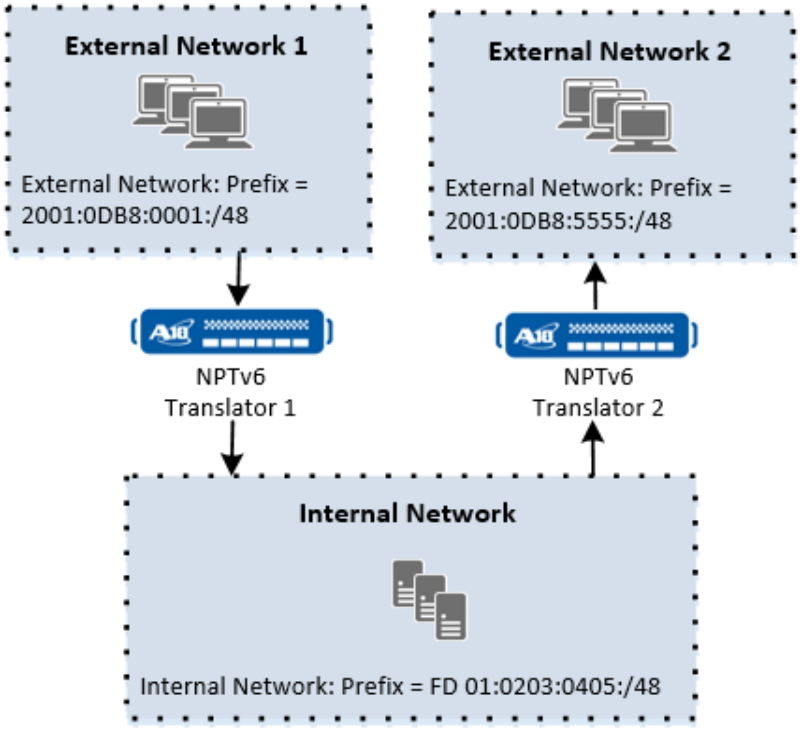
  Translation is required for the source and the destination addresses of the packet.

Figure 37 : Prefix Translation Between Two Private Networks



- Between one internal network and multiple external networks.

Figure 38 : Prefix Translation Between an Internal Network and Multiple External Networks



Configuring NPTv6

You must complete the following tasks before configuring NPTv6:

- Configuring the NPTv6 Domain
- Binding the Domain to an Interface

You can also enter commands to enable the following tasks:

- Enabling or Disabling ICMPv6 Error Notifications
- Displaying NPTv6 Statistics for a Domain
- Clearing NPTv6 Statistics for a Domain
- Displaying an NPTv6 Domain

Configuring NPTv6 by using the CLI

You can configure the NPTv6 domain or bind the domain by using the CLI.

Configuring the NPTv6 Domain

1. From the configuration mode, enter the following command to specify the NPTv6 domain name:

   ```
   cgnv6 nptv6 domain domain-name
   ```

   There is no default value, but you can specify between 1-63 characters.

2. From the domain configuration mode, enter the following command to specify the inside prefix:

   ```
   inside-prefixinside-prefix
   ```

   inside-prefix sets the inside-prefix for an NPTv6 domain.

3. From the domain configuration mode, enter the following command to specify the outside prefix:

   ```
   outside-prefixoutside-prefix
   ```

   outside-prefix sets the outside-prefix for an NPTv6 domain.

NOTE: There is no default value to set the inside and outside prefixes.

When configuring the inside or the outside prefixes, consider the following information:

- Each NPTv6 domain consists of one inside prefix (inside-prefix) and one outside prefix (outside-prefix).
- The maximum prefix length is 64.
- The inside prefix length must be same as the outside prefix length. If the prefixes are of different lengths, a zero (0) is added to the shorter prefix so that both prefixes are of the same length.
- The ACOS device allows a maximum of 8 NPTv6 domains for each virtual network partition (VNP).

Binding the Domain to an Interface

To bind the NPTv6 domain to the relevant interfaces, enter the following commands:

```
interface {ethernet portnum/ethernet ve ve-num/trunk trunk id}
nptv6 domain-name inside/outside
interface {ethernet portnum/ve ve-num/trunk trunk id}
nptv6 trunk id outside
```

Table 19 : Options when Binding the Domain to an Interface

Option	Description
ethernet *portnum*	Specifies the ethernet interface
ve *ve-num*	Specifies the virtual ethernet interface
trunk interface	Specifies the domain name.
inside	Specifies that the inside prefix will be translated
outside	Specifies that the outside prefix will be translated

NOTE: ACOS allows a maximum of 4 NPTv6 domains to bind to an interface.

Enabling or Disabling ICMPv6 Error Notifications

You can enable or disable the ICMPv6 error notification when the packet needs prefix translation, but the translation fails.

To enable or disable ICMPv6 error notification, enter the following command:

```
[no] cgnv6 nptv6 common send-icmpv6-on-error
```

For more information about the types of ICMPv6 errors that might occur, see RFC 4443.

Displaying NPTv6 Statistics for a Domain

Enter the following command to display NPTv6 statistics for a domain:

```
show cgnv6 nptv6 statistics [domain-name]
```

Clearing NPTv6 Statistics for a Domain

Enter the following command to clear NPTv6 statistics for a domain:

```
clear cgnv6 nptv6 statistics [domain-name]
```

Displaying an NPTv6 Domain

Enter the following command to display an NPTv6 domain:

```
show cgnv6 nptv6 domains [domain-name]
```

CLI Examples

The following examples show you how to configure IPv6 prefix translation in different network configurations.

Prefix Translation between One Internal Network and One External Network

This example shows you how to configure IPv6 prefix translation between an internal and an external network.

Traffic moves between the Internal Network (inside prefix `FD01:0203:0405:/48`) and the External Network (external prefix `2001:0DB8:0001:/48`).

The following example is illustrated in this figure:

```
cgnv6 nptv6 domain domain1

inside-prefix FD01:0203:0405:/48

outside-prefix 2001:0DB8:0001:/48

interface ethernet/ve/trunk 1

nptv6 domain 1 inside

interface ethernet/ve/trunk 2

nptv6 domain 1 outside
```

Prefix Translation between Two Private Networks

This example shows you how to configure IPv6 prefix translation between two private networks. One ACOS device acts as the NPTv6 translator between the internal network and the external network.

Traffic moves from Internal Network A (internal prefix `FD01:4444:5555:/48`) to External Network B (external prefix `2001:0DB8:0001:/48`) and then to Internal Network C (internal prefix `FD01:0203:0405:/48`). When the process is reversed, traffic moves from Internal Network C to External Network B and then to Internal Network A.

The following example is illustrated in this figure:

NPTv6 Domain 1

```
inside-prefix FD01:0203:0405:/48

outside-prefix 2001:0DB8:0001:/48
```

NPTv6 Domain 2

```
inside-prefix FD01:4444:5555:/48

outside-prefix 2001:0DB8:6666:/48
```

```
interface ethernet/ve/trunk 1

    nptv6 domain 1 inside

    nptv6 domain 2 outside
```

```
interface ethernet/ve/trunk 2

    nptv6 domain 2 inside

    nptv6 domain 1 outside
```

Prefix Translation between One Internal Network and Multiple External Networks

This example shows you how to configure IPv6 prefix translation between an internal network and multiple external networks. There are two ACOS devices that act as NPTv6 translators, one for each external network.

Traffic moves between the Internal Network (internal prefix `FD01:0203:0405:/48`) to External Network #1 (external prefix `2001:0DB8:0001:/48`) or to External Network #2 (external prefix `2001:0DB8:6666:/48`).

The following example is illustrated in this figure:

NPTv6 Domain 1

```
inside-prefix FD01:0203:0405:/48
outside-prefix 2001:0DB8:0001:/48
```

NPTv6 Domain 2

```
inside-prefix FD01:0203:0405:/48
outside-prefix 2001:0DB8:6666:/48
```

```
interface ethernet/ve/trunk 1
    nptv6 domain 1 inside
    nptv6 domain 2 inside
interface ethernet/ve/trunk 2
    nptv6 domain 1 outside
interface ethernet/ve/trunk 3
    nptv6 domain 2 outside
```

This example is illustrated in this figure.

IPv6 Rapid Deployment (6rd)

This chapter describes IPv6 Rapid Deployment (6rd) and how you can configure it on an ACOS device.

The following topics are covered:

Overview

IPv6 Rapid Deployment (6rd) allows IPv6 clients and IPv6 servers that are separated by IPv4 networks to communicate without changing the IPv4 network.

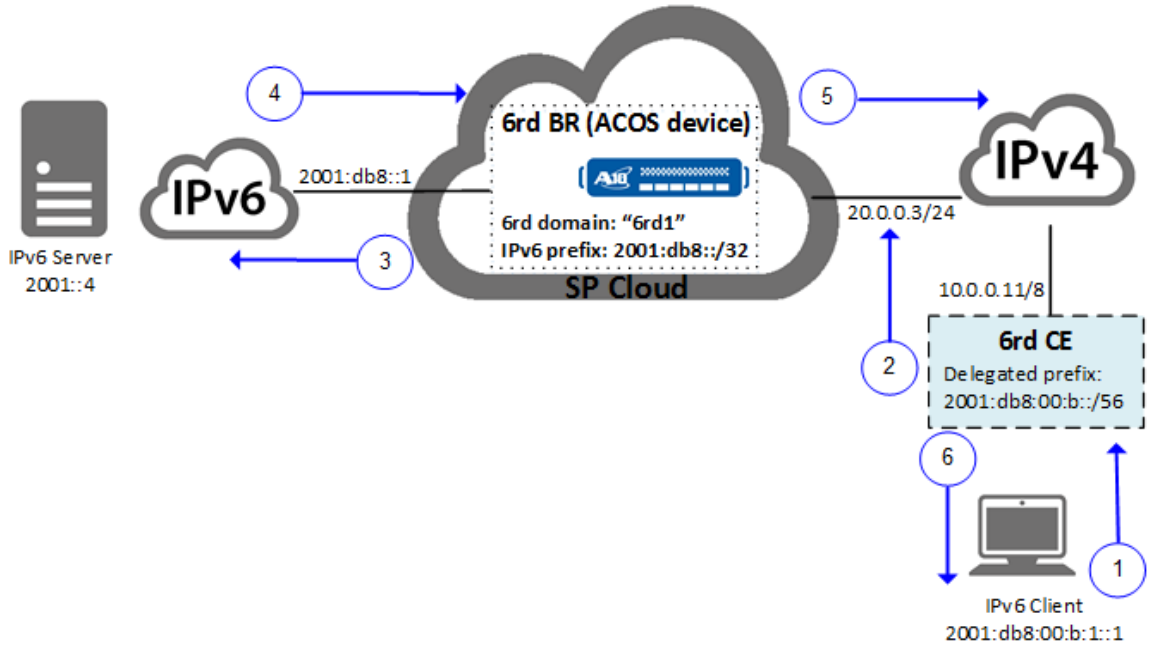
6rd is based on RFX 5969, IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification.

6rd also uses CGN standards for the NAT component. For more information, see Large Scale Network Address Translation. For information about logging, see the Traffic Logging Guide for IPv6 Migration.

To send the IPv6 traffic over IPv4, 6rd uses an IPv4 tunnel. The tunnel origination point on the sender's side of the tunnel encapsulates the IPv6 traffic in IPv4 packets and sends these packets over IPv4 to the device at the remote end of the tunnel. The device at the remote end of the tunnel decapsulates the packets and sends them over the IPv6 network to their destination.

Figure 39 illustrates an example of a 6rd deployment, the traffic flow for an IPv6 client request to an IPv6 server, and the server reply to the client.

Figure 39 : 6rd Example



In this example, a service provider has operational control of 6rd domain **6rd1**, and provides 6rd service to IPv6 clients in this domain.

The following procedure provides the high-level steps for this process:

1. Client 2001:db8:00:b:1::1 sends an IPv6 request to server 2001::4. For information about how 6rd client addresses are formed, see 6rd Prefix and Delegated Prefix.
2. The customer edge (CE) router at the client site encapsulates the IPv6 request in one or more IPv4 packets and sends the request over IPv4 to the ACOS device.

   The ACOS device is configured as the 6rd Border Relay (BR) for the 6rd domain.
3. The 6rd border relay (BR), the ACOS device, decapsulates the request and sends it over the IPv6 network to the IPv6 server.
4. The IPv6 server sends the reply over the IPv6 network.
5. The 6rd BR (ACOS device) encapsulates the reply in one or more IPv4 packets and sends them to the client's CE router.
6. The CE router decapsulates the reply and sends it to the client.

NOTE: For the ACOS BR address, you can use an IP address that is configured on an ACOS interface or a floating-IP address. If you use an IP address that is configured on an ACOS interface, the 6rd domain is not synchronized to the standby ACOS device as part of configuration synchronization.

6rd Prefix and Delegated Prefix

Traffic that belongs to a 6rd domain can be identified by the 6rd prefix. The 6rd prefix consists of a unique value for the high order (leftmost) bits in the IPv6 addresses of 6rd clients. All clients that have a given 6rd prefix belong to the same 6rd domain. In Figure 39, the 6rd prefix is 2001:db8::/32.

When you configure 6rd on the ACOS device, one of the parameters you provide is the 6rd prefix value. Each 6rd domain can have one 6rd prefix. You can configure multiple 6rd domains on the ACOS device, but each domain must have its own unique 6rd prefix.

NOTE: You can have a maximum of eight 6rd domains.

Delegated Prefix

The delegated prefix provides a unique 6rd identifier to each customer site and consists of the 6rd prefix and the host portion of the CE router's IPv4 interface to the 6rd BR.

```
6rd_Prefix:CE_IPv4_Address
```

In this figure, the 6rd delegated prefix for the client is 2001:db8:00:b::/56. The host portion of the CE router's IPv4 address, ".0.0.11", becomes "00:b" in the delegated prefix.

NOTE: The delegated prefix is assigned by another mechanism, such as DHCPv6, and not the ACOS device.

6rd Client Address

The 6rd client's IPv6 address consists of the 6rd prefix, the delegated prefix, a subnet ID, and an interface ID.

```
6rd_Prefix:CE_IPv4_Address:Interface_ID
```

NOTE: The current release does not support use of anycast addresses for 6rd.

Client CE IPv4 Network

If the entire 32-bit CE IP address of the client is not included in the client's 6rd delegated prefix, you must specify the client IPv4 network and the mask length. The mask length indicates the portion of the network address that is the same for all 6rd clients in the domain. In Figure 39, the first 8 bits of the CE router IPv4 address are the same for all clients. Therefore, the client CE IPv4 mask length is 8. Only the 24 lower-order bits of the CE router IPv4 address are used in a client's 6rd delegated prefix.

Packet Fragmentation

The ACOS device uses the following fragmentation settings for 6rd by default:

- Inbound IPv6 packets from IPv6 servers to IPv6 clients – Drops oversize inbound IPv6 packets and sends an ICMPv6 error message back to the server.
- Fragmentation is not performed.
- Outbound IPv6 packets from the ACOS device, forwarded on behalf of 6rd clients to IPv6 servers – Fragments oversize IPv6 packets.
- Don't Fragment bit set in outbound IPv6 packets – Drops oversize outbound IPv6 packets and sends an IPv4 ICMP error message to the client's 6rd CE.

These settings are configurable.

The default maximum transmission unit (MTU) for the IPv6 tunnel is 1480 bytes, which is configurable.

NOTE: Packet virtual reassembly is required for Carrier Grade NAT (CGN) devices to perform NAT and handle ALG traffic.

6rd Interoperability with Other IPv6 Migration Protocols

6rd can interact with other IPv6 migration protocols such as NAT64, DS-Lite.

Consider the following information:

- 6rd and LSN can be enabled concurrently for the same client.

  Non-6rd traffic from the client will receive LSN processing and 6rd traffic will receive 6rd processing.

- DS-Lite traffic that is not terminated on the ACOS device will pass through 6rd even if the traffic matches the DS-Lite class list.
- DS-Lite packets that originate from 6rd are processed correctly at Layer 3 after de-tunneling.
- DS-Lite packets that originate from native IPv6, and whose destination is behind 6rd, will go through 6rd.
- If the source IPv6 address matches the class list and the destination IPv6 address contains the configured NAT64 prefix, after de-tunneling, 6rd traffic is handled as NAT64 traffic.
- Hair-pinning is supported between a 6rd client with a NAT64 mapping and any of the following:
  - Another 6rd client with a NAT64 mapping
  - A NAT64 client
  - An LSN client
- IPv6 traffic whose source address matches a NAT64 class list, and whose destination is in a 6rd domain, is handled as 6rd traffic.
- FTP, TFTP and RTSP ALG are supported for 6rd-NAT64 inter-working. SIP ALG is not supported in the current release.
- Fragmentation and reassembly are supported for 6rd-NAT64 inter-working.

NAT64 rules are used for traffic going to the IPv4 network and 6rd rules are used for traffic going to the 6rd domain.

Support for 6rd inter-working with other IPv6 migration protocols does not require any configuration changes and can not be disabled.

Configuring IPv6 Rapid Deployment (6rd)

Configuring 6rd by Using the GUI

1. Navigate to **CGN > SixRD**.
2. To configure the domain settings, click **Add**.
   a. Enter the name of the domain.
   b. In BR IPv4 Address, enter the 6rd IPv4 address of the ACOS device. The IPv4 address must be one of the following:
      - An IP interface that is already configured on the ACOS device. The interface must be connected to the 6rd domain's clients.
      - A floating-IP interface that is already configured on the ACOS device. In this case, the VRRP-A state is applicable. Packets are forwarded only on the active ACOS device in the VRRP-A pair.
   c. Enter the IPv6 prefix for the 6rd domain.
   d. In Customer Edge IPv4 Network, enter the client IPv4 network and the portion of the client's 6rd CE router IPv4 address that is common to all of the 6rd domain's clients.

      NOTE: If the entire 32-bit CE IP address of the client will be included in the client's 6rd delegated prefix, you can leave the Customer Edge IPv4 Network fields blank. In this case, the mask length is 0.

   e. (Optional) To change the MTU for the IPv6 tunnel, enter the value in the MTU field. You can specify 1280-1480 bytes, and the default is 1480.

f. Under **Action**, click the icon to save the new row or to cancel and hide the row.

g. Repeat for each domain.

3. (Optional) To change 6rd fragmentation settings, click **Add**.

a. Select the traffic direction.

These are the actions for Inbound:

- Drop – Drops oversize packets without sending an ICMPv6 error message back to the server. Fragmentation is not performed.
- IPv4 – The IPv6 packet is treated as an IPv4 payload, and the IPv4 packet is fragmented. The client's 6rd CE router defragments the IPv4 packet, extracts the IPv6 payload, and sends it to the IPv6 client.
- IPv6 – The IPv6 packet is fragmented, and the fragments are placed into separate IPv4 packets. The IPv4 packets are not fragmented. The fragmented IPv6 packet is defragmented by the IPv6 client.
- Send ICMPv6 – Drops oversize packets and sends an ICMPv6 error message back to the server. Fragmentation is not performed.

These are the actions for Outbound:

  - Send ICMP – Drops oversize packets and sends an IPv4 ICMP error message to the client's 6rd CE router. Fragmentation is not performed.
  - Drop – Drops oversize packets without sending an ICMPv6 error message to the client. Fragmentation is not performed.
  - IPv6 – Fragments oversize IPv6 packets.
  - Send ICMPv6 – Drops oversize packets and sends a tunneled ICMPv6 error message to the client. Fragmentation is not performed.

b. To specify the ACOS response to oversize outbound IPv6 packets that have the Don't Fragment bit set, select **df-set** and select one of the following actions:

- Send ICMP – Drops oversize packets and sends an IPv4 ICMP error message to the client's 6rd CE router.

- Drop – Drops oversize packets without sending a tunneled ICMPv6 error message to the client.
- IPv6 – Fragments oversize IPv6 packets anyway and forwards the fragments.
- Send ICMPv6 – Drops oversize packets and sends a tunneled ICMPv6 error message to the client.

4. Click **Update**.

Configuring 6rd by Using the CLI

Configure 6rd Domain Parameters

The following commands configure the 6rd domain in this figure.

1. Enter the following command to configure a 6rd domain:

```
ACOS(config)# cgnv6 sixrd domain 6rd1
ACOS(config-domain:6rd1)#
```

2. Enter the following command to specify the 6rd IPv4 address of the ACOS device and the IPv6 prefix for the 6rd domain:

```
ACOS(config)# interface ethernet 1
ACOS(config-if:ethernet:1)# ip address 1.1.4.1 /24
ACOS(config-if:ethernet:1)# exit
ACOS(config)# interface ethernet 2
ACOS(config-if:ethernet:2)# ipv6 address 2001:db8::/32
ACOS(config-if:ethernet:2)# exit
ACOS(config)# cgnv6 sixrd domain 6rd1
ACOS(config-domain:6rd1)# br-ipv4-address 20.0.0.3 ipv6-prefix
2001:db8::/32
```

NOTE: The IPv4 address must meet one of the following requirements:
- An IP interface that is already configured on the ACOS device. The interface must be connected to the 6rd domain's clients.
- A floating-IP interface that is already configured on the ACOS device. In this case, the High Availability (HA) state is applicable. Packets are forwarded only on the active ACOS device in the HA pair.

3. Enter the following command to specify the client IPv4 network, and the portion of the client's 6rd customer edge (CE) router IPv4 address that is common to all clients of the 6rd domains:

```
ACOS(config-domain:6rd1)# ce-ipv4-network 10.0.0.0 /8
```

For example, if your deployment uses 10.0.0.0/8 for all CE router IPv4 addresses in the 6rd domain, specify **`ce-ipv4-network 10.0.0.0/ 8`**.

NOTE: If the entire 32-bit CE IP address of the client will be included in the client's 6rd delegated prefix, you can omit this command. In this case, the mask length is 0.

4. Enter the following command to specify the Maximum transmission unit (MTU) of the IPv6 tunnel:

```
ACOS(config-domain:6rd1)#  mtu 1280
```

Changing 6rd Fragmentation Settings (Optional)

1. Enter the following command to change fragmentation support for oversize inbound IPv6 packets. The following example drops oversize packets without sending an IPCMPv6 error back to the server.

```
ACOS(config)# cgnv6 sixrd fragmentation inbound drop
```

NOTE: For packets larger than 1500 bytes, the **`ipv4`** option does not work. In this case, **`ipv6`** is recommended instead.

2. Enter the following command to change fragmentation support for oversize outbound IPv6 packets. The following example fragments oversize IPv6 packets.

```
ACOS(config)# cgnv6 sixrd fragmentation outbound ipv6
```

3. Enter the following command to change the ACOS response to oversize outbound IPv6 packets that have the Don't Fragment bit set. The following example drops oversize packets and sends an IPv4 ICMP error message to the client's 6rd CE router.

```
ACOS(config)# cgnv6 sixrd fragmentation outbound df-set send-icmp
```

Displaying and Clearing 6rd Statistics

1. Enter the following command to display 6rd statistics:

```
ACOS# show cgnv6 sixrd statistics 6rd1
sixrd statistics for domain 6rd1
-----------------------------
Outbound TCP packets received            0
Outbound UDP packets received            0
Outbound ICMP packets received           0
Outbound other packets received          0
Outbound packets dropped                 0
Outbound IPv6 destination unreachable    0
Outbound Fragmented IPv6                 0
Inbound TCP packets received             0
Inbound UDP packets received             0
Inbound ICMP packets received            0
Inbound other packets received           0
Inbound packets dropped                  0
Inbound IPv4 destination unreachable     0
Inbound Fragmented IPv4                  0
Inbound Fragmented IPv6 in tunnel        0
Traffic match SLB virtual port           0
Unknown sixrd delegated prefix           0
Packet too big                           0
Not local IP                             0
Fragment processing errors               0
Other errors                             0
```

2. Enter the following command to clear 6rd statistics:

```
ACOS# clear cgnv6 sixrd statistics6rd1
```

Hardware Offloading Of CGN Sessions

This chapter describes Hardware Offloading of CGN sessions and how you can configure it on an ACOS device.

The following topics are covered:

Overview

In general, all CGN sessions are processed and forwarded using the software. This may cause a spike in CPU usage in case of heavy network traffic. In such scenarios, you can offload some CGN sessions to the hardware. Hardware offloading frees up the CPU cycles, thereby enhancing the overall throughput, efficiency, and latency of the Thunder device. This feature can be enabled by configuring the **`system hardware-accelerate session-forwarding`** command.

In ACOS, hardware offloading (also known as hardware acceleration or hardware forwarding) is only supported on FPGA devices having Ternary Content-Addressable Memory (TCAM). TCAM is a specialized memory used in high-speed search applications. Thunder devices having a TCAM provide the capability to forward CGN sessions to the hardware by programming the flows in the available TCAM.

The packet processing statistics are periodically exported to the software. These statistics can be viewed using show commands (see CLI Configuration).

Hardware Offloading Features in ACOS

Following are the features of hardware offloading in ACOS:

- Supported only on TH7655 (Lite), TH6435, and TH14045 devices.
- Only the following sessions can be offloaded to the hardware:
  - CGN LSN (NAT44) and NAT64
  - Fixed NAT
  - Firewall (Only TCP and UDP traffic)
- Up to 128K IPv4 sessions or 64K IPv6 sessions can be offloaded to the hardware.
- Only established sessions can be offloaded to hardware.
- Only active sessions can be forwarded; standby sessions cannot be forwarded.

CLI Configuration

- To enable hardware offloading of CGN sessions, configure the following command:

```
ACOS(config)# system hardware-accelerate session-forwarding
```

- To check if a particular session is forwarded using hardware, use the **show sessions** command. The flags field in the show command output indicates that the session is offloaded to the hardware.
- To view the information on the hardware entries being programmed into the TCAM, use the **show hardware-accelerate statistics** command.

  Consider the following show command output:

```
ACOS# show hardware-accelerate statistics
Security Policy Engine Traffic statistics
-----------------------------------------
Total Hit Counts                          5
Total IPv4 hardware Hit Counts            5
Total IPv6 hardware Hit Counts            0
Total Available IPv4 SPE Entries          131072
Total Available IPv6 SPE Entries          65536
Total IPv6 hardware forwarded packets     0
Total SPE programming requests            8
Total SPE programming errors              0
Total Ageeout Drop Counts                 0
Total Flow singlebit Errors               0
```

```
Total Tag Mismatch Errors                    0
Total Sequence Mismatch Errors               0
Total Program Invalidation drop Counts       0
Total Flow Drop Counts                       0
Total Flow Error Counts                      0
Total Flow Unalign Counts                    0
Total Flow Underflow Counts                  0
Total Flow TX Full Drop Counts               0
Total Flow QDR Full Drop Counts              0
Total Flow Phyport Mismatch Drop Counts      0
Total Flow VLAN-ID Mismatch Drop Counts      0
Total Flow VMID Mismatch Drop Counts         0
Total Flow Protocol Mismatch Drop Counts     0
```

The **`Total IPv4 hardware Hit Counts`** field indicates that five IPv4 packets are hardware forwarded. Similarly, the **`Total SPE programming requests`** field indicates that eight SPE programming requests are made to forward packets to the hardware.

- To view the hardware offloading statistics for CGN sessions, use the following show commands:
  - For LSN (NAT44) sessions - **`show counters cgnv6 lsn hw-accelerate`**
  - For NAT64 sessions - **`show counters cgnv6 nat64 hw-accelerate`**
  - For Fixed-NAT sessions - **`show counters cgnv6 fixed-nat hw-accelerate`**

  Example output for **`show counters cgnv6 lsn hw-accelerate`** command:

```
ACOS# show counters cgnv6 lsn hw-accelerate
show counters cgnv6 lsn hw-accelerate
************************************
HW Entries Created                                4
HW Entry Creation Failed                          0
HW Entry Creation Failed - server down            0
HW Entry Creation Failed - max entries exceeded   0
HW Entries Freed                                  4
HW Entries Freed - opposite tuple entry aged-out  2
HW Entry Freed - no HW prog                       0
HW Entry Freed - no matched conn                  0
HW Entry Freed - no software entry                16
```

```
HW Entries Count                                0
HW Entries Aged Out                             8
HW Entries Aged Out - idle timeout              8
HW Entries Aged Out - TCP FIN                   0
HW Entries Aged Out - TCP RST                   0
HW Entries Aged Out - invalid dst               0
HW Entries Force HW Invalidate                  0
HW Entries Invalidate due to server down        0
TCAM Flows Created                              2
TCAM Flows Freed                                2
TCAM Flow Count                                 0
```

The output for **show counters cgnv6 nat64 hw-accelerate** and **show counters cgnv6 fixed-nat hw-accelerate** commands is similar to above example output. For field descriptions and more information on command usage, refer to the *Command Line Interface Reference* guide.

Limitations

The following are the limitations of hardware offloading:

- Hardware offloading is not supported for features that involve inspection of the payload. Therefore, the following features cannot be offloaded:
  - ALG sessions (for CGN and Firewall)
  - Tunneled traffic (for example, DS-lite)
  - DNS64 sessions
  - SCTP, GTP traffic
  - Rate-limiting
  - Application classification
  - Stateful and Generic Firewall sessions
  - Firewall local connections
- Since sessions can be offloaded only after they are established, the first few packets are always forwarded to the software.

- Hardware offloaded sessions do not support Round-Robin as a load-balancing method across trunk members.
- Fat flows are not supported.

CGN Compliant RFCs

This chapter provides the details about the RFCs that CGN is compliant with.

The following is the list of RFCs that CGN is compliant with:

Table 20 : List of CGN Compliant RFCs

CGN Requirements	RFCs
LSN	draft-nishitani-cgn-02
NAT for Unicast UDP	RFC 4787
NAT for TCP	RFC 5382
NAT for ICMP	RFC 5508
Port Control Protocol (PCP)	RFC 6887
Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers	RFC 6146
DNS for NAT from IPv6 to IPv4	RFC 6147
EPRT to PORT	RFC 2428
EPSV to PASV	RFC 2428
LPRT to PORT	RFC 1639
LPSV to PASV	RFC 1639
Lw4o6	draft-cui-software-b4-translated-ds-lite-07
DS-Lite	RFC 6333
Stateless NAT46	draft-liu-behavenat46-02
6rd	RFC 5969
NPTv6	RFC 6296

Glossary

A

ARP

Address Resolution Protocol. A communication protocol used for finding the link layer address or a MAC address within a given IP address. Its mapping function is used primarily on IPv4 addresses.

C

CGN

Carrier Grade NAT. An approach of designing IPv4 network in which private network addresses are used for configuring end sites and residential networks. The private network addresses are translated to public IPv4 addresses through an intermediate network address translator embedded in the network.

cluster

A set of distinct or closely-connected computers working in tandem as a single system.

cluster node

A node that can communicate with systems within a cluster for fulfilling a common goal. Cluster nodes can added or removed at any time.

E

EIF

Endpoint-Independent Filtering. A filtering process protocol that checks the destination IP and the destination port of an inbound packet transmitted by an External Endpoint. It determines whether or not to pass the packet.

EIM

Endpoint Independent Mapping. A mapping process protocol that ensures the assignment of same external address and port for connections linked to a given host, provided that they use the same internal port.

F

failover

A backup operational mode that allows the functions of a system component such as a network, server or database to be assumed by secondary components in instances when the primary components are unavailable due to failure or downtime.

G

Gi

GPRS interface. A interface powered by General Packet Radio Service and located between the external Public Data Network and the Gateway GPRS Support Node.

H

hashing

The process of encrypting a version of an IP address with the same IP hash.

I

IP NAT Pool

Internet Protocol Network Address Translation Pool. The process of randomly assigning public IPs from a pool of IPs to private internal IPs based on first come-first serve method.

IP route aggregation

An alternative to route summarization, where the number of routing tables used in an IP network is minimized.

L

L2

A Data Link Layer, the second layer in the seven-layered OSI reference model used for designing network protocols. It consists MAC address, frame relay, token ring and ethernet.

L3V

Layer 3 Virtualization. A virtualization layer that allows organizations to utilize the same IP address ranges for ensuring that the multi-tenant data center architecture gets the flexibility similar to that of a independently-deployed device.

LSN

Large Scale NAT. An approach of designing IPv4 network in which private network addresses are used for configuring end sites and residential networks. The private network addresses are translated to public IPv4 addresses through an intermediate network address translator embedded in the network.

M

MAP-E

Mapping of Address and Port with Encapsulation. A mechanism for IPv6 transition mechanism where IPv4 packets are transported over an IPv6 network by using IP encapsulation. It allows ISPs deliver IPv4 services in the absence of full dual-stack network deployment.

MAP-T

"Mapping of Address and Port using Translation. A mechanism where double translation of IPv4 to IPv6 and vice versa is performed on CE devices and endpoint routers. "

multiple-node

Two or more independent server nodes sharing one or more power resources and the same enclosure.

N

NAT

Network Address Translation. A method of reallocating one IP address space into another by changing the network address information in the IP header when the packets are still being transmitted across a traffic routing device.

P

PCP

Port Control Protocol. A computer networking protocol enabling hosts on IPv4/IPv6 networks a control on the translation of incoming packets being forwarded by an upstream router where network address translation or packet filtering is performed.

R

RADIUS

Remote Authentication Dial-In User Service. A networking protocol that provides centralized AAA management for users connecting and using a network service.

return traffic

Data traffic with a specific route entry that directs packets to another path from the static route set.

S

SCTP

Stream Control Transmission Protocol. A protocol used for transmitting multiple data streams simultaneously between two end-points established

over a connection in a network.

service template

A model of specific values designed for the sole purpose of configuring a service.

SLB

The process of distributing high-traffic sites among multiple servers by using a network-based hardware or software-based device. SLB intercepts the traffic for a website and reroutes it to different servers for attaining data-load equilibrium.

SNMP

Simple Network Management Protocol. A standard Internet Protocol used for collecting and managing information on managed devices over IP networks and for changing the information to modify device behavior.

Source NAT Pool

A pool of source NAT protocols used when an internal host begins a session with an external host and a dual NAT without using the switch IP.

Static-NAT

A one-to-one mapping of private IPs to public IPs when a network device inside a private network is to be accessible from over the internet.

subnet

An IP network subdivision.

Contact Us