Deployment Guide

Carrier Grade NAT (CGN) / Large Scale NAT (LSN)

TABLE OF CONTENTS

1 INTRODUCTION AND SCOPE

In 2011, the Internet Assigned Numbers Authority (IANA) issued the last remaining Class A (/8) address blocks to the Regional Internet Registries (RIR), leaving the RIRs in control of assigning the remainder of the available IPv4 addresses. This increases the difficulty for Internet Service Providers (ISPs) to continue to obtain unallocated IPv4 address space, forcing a plan of action both to preserve the remaining IPv4 address space and to provide a mechanism for IPv6 translation. Many technologies have emerged to solve this problem, including NAT444, DS-Lite, and 6rd; all of which are based upon a common foundation of Carrier Grade Network Address Translation (CGN).

This guide provides a basis for understanding A10 Networks' CGN implementation, and includes an overview of the solution, design and scaling considerations, and an explanation of system configuration, optional features, and traffic logging.

Note: *CGN also is sometimes called Large Scale NAT (LSN), and is the term used in the IETF documents referenced in this document.*

2 CGN OVERVIEW

CGN provides a methodology for preserving IPv4 addresses by centralizing the public address resources and sharing those resources across a large user community. CGN offers the following advantages over traditional NAT operations:

- **High Transparency**
  CGN implements several features to provide a seamless user experience across a NAT environment, including Endpoint-independent Mapping (EIM), Endpoint-independent Filtering (EIF), address pooling, hairpinning, and port preservation. These features provide a transparent client access environment to outside resources, thus insuring that both client-server and peer-to-peer applications continue to function as designed.

- **Well-Defined Behavior**
  CGN is a mature technology whose operation is well standardized by several IETF RFCs and draft documents, including the following:

  - ♦ BEHAVE-TCP (RFC 5382)
  - ♦ BEHAVE-UDP (RFC 4787)
  - ♦ BEHAVE-ICMP (RFC 5508)
  - ♦ CGN (draft-nishitani-cgn-05)

  These RFCs provide a foundation for application transparency and they formalize CGN behavior to facilitate future application development.

- **Fairness and Resource Sharing**
  A10 Networks' CGN implementation provides limits at both session and user levels in order to control the amount of allocated resources. This ensures that resources are distributed fairly across the user-base in accordance with the service provider's requirements.

- **Log File Size Management**
  CGN implementations can create large amounts of logging data in service provider networks. A10 Networks' implementation provides many logging techniques to limit both the number of log entries and their size.

CGN general architecture consists of an access network (addressed with RFC 6598 reserved address 100.64.0.0/10), an aggregation routing layer, CGN devices, and peering routers egressing to the public Internet. For business or residential customers that are directly connected to the access network, there is only one level of NAT (NAT44) required. These customers receive an address directly from the 100.64.0.0/10 subnet. Typically, residential customers deploy a gateway device that implements NAT, creating the NAT444 model. The clients use private addresses from the RFC 1918 IP address space. The private addresses are translated into addresses in the 100.64.0.0/10 subnet, which is configured within the ISP access infrastructure. Client (end-user) traffic then is routed through an aggregation layer to the assigned CGN device, and then translated into IPv4 public addresses. CGN deployment is transparent to end-users and requires no configuration changes to customer-premise equipment (CPE) or hosts.

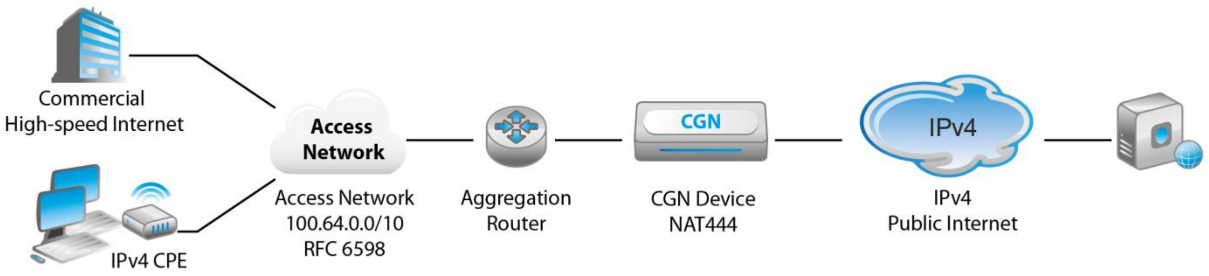


Figure 1: NAT444 high-level architecture

3 SCALABILITY CONSIDERATIONS

The following considerations should be taken into account to ensure CGN scalability meets the service provider's business needs:

- Session scalability
- TCP connection setup rate
- Number of pooled public IP addresses required
- Device throughput

A10 Networks' AX Series provides excellent scalability in a low-cost, small form-factor device, delivering the performance and scale required for CGN deployments in large ISP networks. The AX Series leads the industry in throughput, session scalability, and connection-setup rate performance, and it is available in several versions to ensure CGN sizing at the appropriate performance-price point.

4 BASE CONFIGURATION

This section presents a typical CGN topology, and includes detailed configuration for physical network attributes, High Availability, route integration and redistribution, and address translation using both dynamic and fixed NAT.

4.1 REFERENCE TOPOLOGY

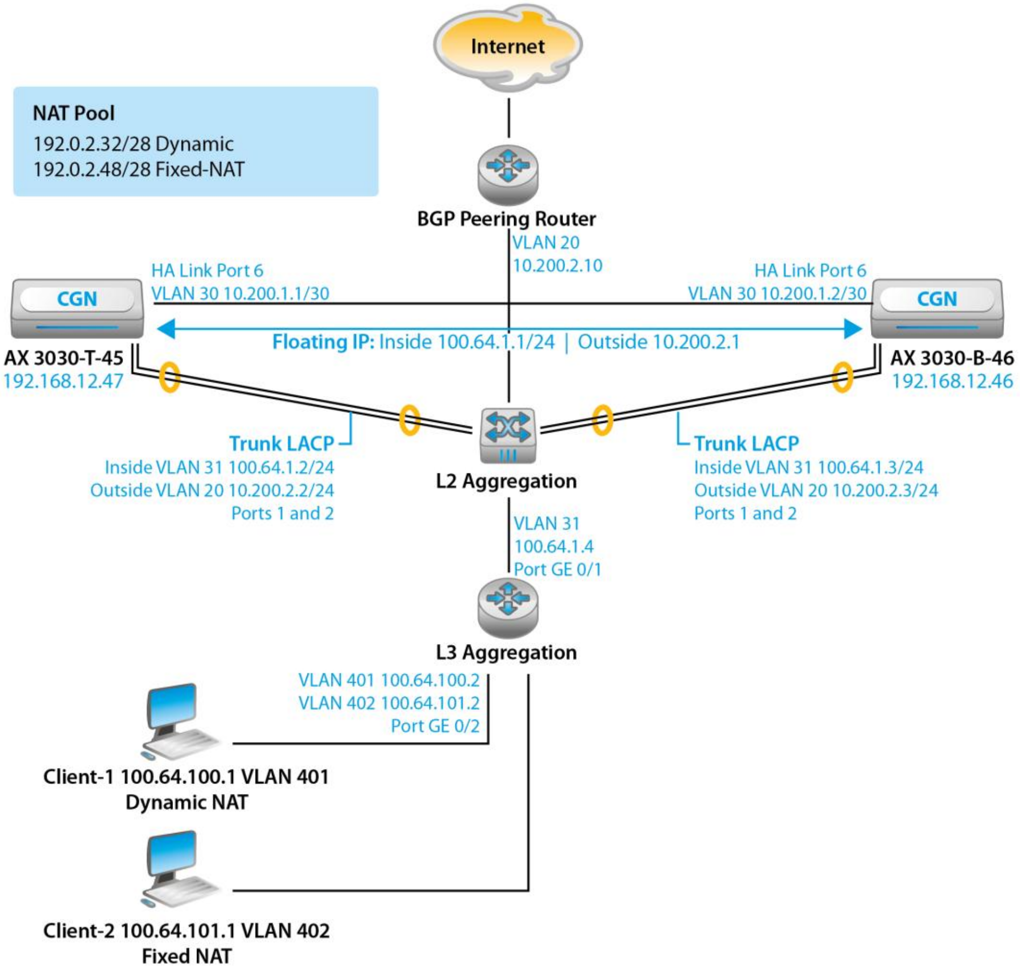


Figure 2: CGN reference topology

The configuration example depicted in Figure 2 illustrates a NAT44 deployment and consists of two Windows clients directly connected to the service provider’s network without a CPE NAT router. Therefore, each client receives an address from the reserved 100.64.0.0/10 subnet. Client-1 is configured for dynamic CGN mapping, and Client-2 uses Fixed-NAT mapping. The aggregation router and Layer 2 switch are connected to two AX devices running in High Availability (HA) mode using LACP link aggregation. A dedicated HA link is utilized for clarity and simplicity. However, the HA protocol also can use the LACP connections. Finally, the BGP peering router is connected to the aggregation router, providing the connection to the Internet.

This example uses dynamic routing protocols to redistribute the NAT pool and the floating IP address. OSPF is used between the aggregation router and the AX device; BGP is configured between the BGP peering router and the AX device. The BGP peering router injects a default route towards the AX device and the AX device injects the configured NAT pool subnets, modifying the next hop to the outside floating IP address 10.200.2.1. The AX device also injects a default route (using OSPF) towards the aggregation router. The next hop must be modified to represent the inside floating IP 100.64.1.1. Finally, the aggregation router must use a policy to redirect the non-translated traffic to the AX device's inside floating IP address.

To further clarify, here is a packet walkthrough of the topology:

1. Client-1 generates a TCP-SYN packet and sends it to the aggregation router, 100.64.100.2.
2. The aggregation router uses a policy to redirect the packet to the AX device's HA floating IP address, 10.64.1.1 on inside VLAN 31.
3. The AX device receives the packet and finds a match in the class list configured for dynamic mapping. The AX device creates a NAT binding and replaces the source address with one that is selected from the NAT pool.
4. The AX device sends the packet to a BGP peering router over outside VLAN 20, and the packet is then forwarded to its destination.
5. The destination returns a SYN-ACK to the BGP peering router. This router has a BGP route to the NAT pool subnet's next hop floating IP address, 10.200.2.1 in VLAN 20.
6. The AX device receives the packet, consults the NAT bindings, replaces the destination address with that of Client-1, and routes the packet towards the aggregation router on inside VLAN 31.
7. The aggregation router sends the packet to Client-1.

4.2 INTERFACE CONFIGURATION

Follow the steps below to configure the AX device's interfaces according to Figure 2:

1. Configure the management interface at the global configuration level using the following commands:

```
login as:admin
Using keyboard-interactive authentication.
Password:********
Last login: Sat Jul 21 06:07:07 2012 from 192.168.52.194

AX system is ready now.

[type ? for help]

AX>enable
Password:********
AX#config
AX(config)#interface management
AX(config-if:management)#ip address 192.168.12.47 255.255.255.0
AX(config-if:management)#ip default-gateway 192.168.12.1
AX(config-if:management)#ip control-apps-use-mgmt-port
```

Note: *To enable management services for Ethernet ports, use the* ***enable-management*** *command. Refer to the "AX Series System Configuration and Administration Guide" for more information.*

2. Set the physical interface attributes. For this example, link aggregation is used.

```
AX(config-if:management)#interface ethernet 1
AX(config-if:ethernet1)#lacp trunk 1 mode active
AX(config-if:ethernet1)#lacp timeout long
AX(config-if:ethernet1)#interface ethernet 2
AX(config-if:ethernet2)#lacp trunk 1 mode active
AX(config-if:ethernet2)#lacp timeout long
```

Note: *Interfaces default to the disabled state. To enable an interface, use the* ***enable*** *command at the configuration level for the interface.*

3. Configure VLAN assignments and assign IP addresses to the loopback and virtual interfaces.

   At the configuration level for each VLAN, specify the interfaces to include in the VLAN, add a description, and add a virtual router interface. This example requires three VLANs:

   - Inside (access network to CGN)
   - Outside (public Internet)
   - Inter-chassis link for HA

Note: *IP addresses can be assigned directly to individual Ethernet ports. However, assignment to virtual interfaces allows more flexibility and eases future configuration modifications.*

```
AX(config-if:ethernet2)#vlan 20
AX(config-vlan:20)#tagged ethernet 1 to 2
AX(config-vlan:20)#router-interface ve 20
AX(config-vlan:20)#name "outside"
AX(config-vlan:20)#vlan 30
AX(config-vlan:30)#untagged ethernet 6
AX(config-vlan:30)#router-interface ve 30
AX(config-vlan:30)#name "HA_LINK"
AX(config-vlan:30)#vlan 31
AX(config-vlan:31)#tagged ethernet 1 to 2
AX(config-vlan:31)#router-interface ve 31
AX(config-vlan:31)#name "inside"
```

4. Assign an IP address to the virtual router interface of each VLAN.

```
AX(config-vlan:31)#interface ve 20
AX(config-if:ve20)#ip address 10.200.2.2 255.255.255.0
AX(config-if:ve20)#ip nat outside
AX(config-if:ve20)#interface ve 30
AX(config-if:ve30)#ip address 10.200.1.1 255.255.255.252
AX(config-if:ve30)#interface ve 31
AX(config-if:ve31)#ip address 100.64.1.2 255.255.255.0
AX(config-if:ve31)#ip nat inside
AX(config-if:ve31)#interface loopback 1
AX(config-if:loopback1)#ip address 17.17.17.17 255.255.255.255
AX(config-if:loopback1)#end
```

5. Use the following commands to verify the interface configuration:

   - **show interfaces brief**
   - **show vlans**
   - **show trunk**
   - **show ip interfaces**

   Output examples for each command are shown below.

To verify that the interfaces are up:

```
AX#show interfaces brief
Port  Link  Dupl  Speed Trunk Vlan MAC             IP Address      IPs  Name
--------------------------------------------------------------------------------
mgmt  Up    Full  100   N/A   N/A  000d.480a.69c1  192.168.12.47/24  1
1     Up    Full  1000  1     Tag  000d.480a.69cb  0.0.0.0/0         0
2     Up    Full  1000  1     Tag  000d.480a.69ca  0.0.0.0/0         0
3     Disb  None  None  None  1    000d.480a.69c9  0.0.0.0/0         0
4     Disb  None  None  None  1    000d.480a.69c8  0.0.0.0/0         0
5     Disb  None  None  None  1    000d.480a.69c7  0.0.0.0/0         0
6     Up    Full  1000  None  30   000d.480a.69c6  0.0.0.0/0         0
7     Disb  None  None  None  1    000d.480a.69c5  0.0.0.0/0         0
8     Disb  None  None  None  1    000d.480a.69c4  0.0.0.0/0         0
9     Disb  None  None  None  1    000d.480a.69c3  0.0.0.0/0         0
10    Disb  None  None  None  1    000d.480a.69c2  0.0.0.0/0         0
ve20  Up    N/A   N/A   N/A   20   000d.480a.69cb  10.200.2.2/24     1
ve30  Up    N/A   N/A   N/A   30   000d.480a.69cb  10.200.1.1/30     1
ve31  Up    N/A   N/A   N/A   31   000d.480a.69ca  100.64.1.2/24     1
lo1   Up    N/A   N/A   N/A   N/A  N/A             17.17.17.17/32    1
```

To verify the VLAN configuration:

```
AX#show vlans
Total VLANs: 4
VLAN 1, Name [DEFAULT VLAN]:
  Untagged Ports:     3    4    5    7    8    9   10
    Tagged Ports:    None

VLAN 20, Name [None]:
  Untagged Ports:    None
```

```
    Tagged Ports:     1    2
 Router Interface: ve 20

VLAN 30, Name [HA_LINK]:
  Untagged Ports:     6
    Tagged Ports:   None
 Router Interface: ve 30

VLAN 31, Name [None]:
  Untagged Ports:   None
    Tagged Ports:     1    2
 Router Interface: ve 31
```

To verify link aggregation:

```
AX#show trunk
Trunk ID         : 1       Member Count: 2
Trunk Status     : Up
Trunk Type       : Dynamic (LACP)
Admin Key        : 1001
Members          : 1   2
Cfg Status       : Enb Enb
Oper Status      : Up  Up
Ports-Threshold  : None
Working Lead     : 1
```

To verify the IP address assignments on the VLAN virtual interfaces:

```
AX#show ip interfaces
Port    IP              Netmask         PrimaryIP   Name
----------------------------------------------------------------------------
mgm     192.168.12.47   255.255.255.0   Yes
ve20    10.200.2.2      255.255.255.0   Yes
ve30    10.200.1.1      255.255.255.252 Yes
ve31    100.64.1.2      255.255.255.0   Yes
lo1     17.17.17.17     255.255.255.255 Yes
```

4.3 HIGH AVAILABILITY CONFIGURATION

High Availability (HA) is an AX feature that provides device-level redundancy, thus ensuring continuity of CGN service to clients. In HA configurations, AX devices are deployed in pairs. If the active AX device in the HA pair becomes unavailable, the other AX device assumes the active role and operations continue normally. The following items need to be configured to enable HA operation.

- HA ID – The HA ID of AX1 is 1 and the HA ID of AX2 is 2. Each AX device in an HA deployment must have a unique HA ID.
- HA group – HA group 1 is configured on each AX device. An AX device can have up to 31 HA groups. Each HA group must be configured with a priority. The priority can be used as a tiebreaker for active device selection. Each HA group has a shared MAC address, 021f.a0000.00*xx*. The 02 portion of the address indicates this is an HA virtual MAC address, instead of a system MAC address (00). The *xx* portion of the address is unique to the HA group. The shared MAC address is used for all IP addresses for which HA is provided, including source NAT addresses and floating IP addresses.
- Interfaces – The interfaces associated with the HA instance.
- Floating IP addresses – The floating IP address should be used by downstream devices as their default gateway. The same floating IP address is shared by both AX devices in the HA pair. Regardless of which device is active, downstream devices can reach their default gateway at this IP address.
- Session synchronization – Also called connection mirroring, session synchronization sends information about active CGN sessions to the standby AX device. If a failover occurs, the CGN sessions are maintained without interruption.

4.3.1 HA CONFIGURATION STEPS

To configure the HA items described above, use the following steps.

1. Set the HA identifier at the global configuration level. The HA ID must be different on the active and standby devices. Also specify the group and priority for the HA instance. The group ID must be the same across the HA pair.

```
AX#config
AX(config)#ha id 1
AX(config)#ha group 1 priority 102
```

2. Specify the interfaces that participate in the HA group. By default, the interface sends HA heartbeats. Use the **no-heartbeat** option for interfaces where the heartbeat should not be active.

```
AX(config)#ha interface ethernet 1 no-heartbeat
AX(config)#ha interface ethernet 6
```

3. Configure connection mirroring to synchronize CGN session data across the HA pair. If preemption is enabled, configuration changes to either the HA priority or HA ID will cause a transition to Standby.

```
AX(config)#ha conn-mirror ip 10.200.1.2
AX(config)#ha preemption-enable
```

4. Specify the floating IP addresses for the HA group.

```
AX(config)#floating-ip 10.200.2.1 ha-group 1
AX(config)#floating-ip 100.64.1.1 ha-group 1
```

5. Verify proper HA operation using the following commands:

   - **show ha detail**
   - **show ha statistics**

   Output examples for each command are shown below.

To show detailed HA information:

```
AX(config)#show ha detail
Local Unit:     UP              Peer Unit:      UP
HA Group        Unit            State           Priority
1               Local           Active          102
                Peer            Standby         101

HA Group                        Active          Standby
1               Transitions     5               5

Connectivity:        Server Ports          0         Router Ports          0
HA packets:          Sent                  3815346   Received
3815021
Conn Sync:           Sent                  4830667   Received              7951
Conn Query:          Sent                  282       Received              3796

Conn Sync Create Session:         Sent                    718009    Received
1343
Conn Sync Update Age:             Sent                    3647974   Received
6144
Conn Sync Del Session:            Sent                    715528    Received
935
```

```
Conn Sync Create Persist Sess:   Sent                 0         Received
0
Conn Sync Update Persist Age:    Sent                 0         Received
0
Conn Sync Del Persist Session:   Sent                 0         Received
0

HA errors:
Dup HA ID            0         Invalid Group        0
Version Mismatch     0         SetId Mismatch       0
Missed Heartbeat     1         Timer Msgs           0

HA Port         Sent            Recvd           Missed Heartbeat Backup
Triggered Backup Stopped
1               0               0               0               1
0
6               3815346         3815021         1               1
1
```

To show HA statistics:

```
AX(config)#show ha statistics
Session Sync Packets             Sent      Received
-----------------------------------------------------
Conn Sync:                       4831579   7951
Conn Query:                      282       3796

Session Sync Msg                 Sent      Received
-----------------------------------------------------
Conn Sync Create Session:        718147    1343
Conn Sync Update Age:            3648614   6144
Conn Sync Del Session:           715677    935
Conn Sync Update Seq Num:        0         0
Conn Sync Create with Ext:       0         0

Session Sync Send Errors:
-------------------------
Conn Sync Get Buff Failure:      0
Conn Sync Invalid Interface:     0
```

```
Session Sync Receive Errors:
----------------------------
Conn Sync Create Conn Exists:    2
Conn Sync Del Conn not Found:    55
Conn Sync Virt Port Not Found:   0
Conn Sync Real Port Not Found:   0
Conn Sync Get Conn Failure:      0
Conn Sync Proc Ext Bit Failure:  0
Conn Sync App Type Invalid:      0
Conn Sync Protocol Invalid:      0
Conn Sync Length Invalid:        0
Conn Sync Unknown Type:          0
Conn Sync Packet Empty:          0

Session Query Send Errors:
--------------------------
Conn Query Get Buff Failure:     0
Conn Query Invalid Interface:    0

Session Query Receive Errors:
-----------------------------
Conn Query Unknown Type:         0
Conn Query Packet Empty:         0

Session Sync Profiling Info:
----------------------------

                     Max Sync Msg Per Pkt Sent Min Sync Msg Per Pkt Sent
----------------------------------------------------------------------
Data CPU 1           8                         1
Data CPU 2           8                         1
Data CPU 3           7                         1
Data CPU 4           8                         1
Data CPU 5           7                         1

                     Max Query Msg Per Pkt Sent Min Query Msg Per Pkt Sent
----------------------------------------------------------------------
Data CPU 1           1                         1
Data CPU 2           1                         1
Data CPU 3           1                         1
Data CPU 4            1                          1
Data CPU 5            1                          1
```

4.4 CGN CONFIGURATION

This section focuses on the CGN configuration. The following configuration steps enable CGN.

- Configure NAT pools (and optionally, pool groups). Use the **lsn** option to indicate that the pools are for use by the CGN feature. (This is shown in the syntax example.)
- Configure CGN Limit IDs (LIDs). For each LID, specify the NAT pool to be used. Optionally, set user quotas for the LID.
- Configure class lists for the user subnets that require CGN. A class list is a list of internal subnets or hosts. Within a class list, you can bind each internal subnet to an individual CGN LID.
- Bind a class list to the CGN feature. The class list will apply to packets from the inside NAT interface to the outside NAT interface. There can be at most one class list for this purpose.
- Enable inside NAT on the interface connected to the internal clients.
- Enable outside NAT on the interface connected to the Internet.

4.4.1 CGN CONFIGURATION STEPS

To configure the CGN items described above, use the following steps.

1. Configure NAT pools with the following command at the global configuration level. You must declare a pool name, the range of IP addresses to be used for NAT, and the netmask. Since HA is configured, the **ha-group** option also must be included.

   ```
   AX(config)#ip nat pool cgn-dynamic 192.0.2.33 192.0.2.46 netmask /28
   ha-group-id 1 lsn
   ```

   Alternatively, NAT pools can be combined into pool groups. This simplifies future changes to the configuration and allows non-contiguous address bundling. Use the following command to create a pool group. Declare a pool group name (“example”) and list NAT pools to be included in the group (“cgn-dynamic”).

   ```
   AX(config)#ip nat pool-group example cgn-dynamic ha-group-id 1
   ```

2. Create the CGN Limit ID (LID). The LID associates the NAT pool or pool groups with specific configuration options, including user-quota, override, and rule-lists. The operator can specify up

to 1024 LIDS. Begin the configuration by assigning an LID number. This enters the LSN-LID configuration level.

```
AX(config)#lsn-lid 1
AX(config-lsn lid)#
```

3. Specify the NAT pool or pool groups to be assigned to this LID.

```
AX(config-lsn lid)#source-nat-pool cgn-dynamic
```

4. Specify optional parameters for this NAT pool (See the Advanced Configuration Options section for more details.)

```
AX(config-lsn lid)#user-quota icmp 50
AX(config-lsn lid)#user-quota udp 250 reserve 0
AX(config-lsn lid)#user-quota tcp 250 reserve 0
AX(config-lsn lid)#exit
```

5. Create the class list specifying the internal subnets and hosts that will be associated with a specific LID. In this example, the class list named “vm_client_cgn01” contains a single host 100.64.100.1 and is tied to the configuration in LID 1.

```
AX(config)#class-list vm_client_cgn01
AX(config-class list)#100.64.100.1 /32 lsn-lid 1
AX(config-class list)#exit
```

6. Bind the class list to the CGN process.

```
AX(config)#ip nat inside source class-list vm_client_cgn01
```

7. Declare interfaces for NAT operation. NAT inside is configured for client-side interfaces, while NAT outside is configured for interfaces that are connected to the public Internet.

```
AX(config)#interface ve 20
AX(config-if:ve20)#ip address 10.200.2.3 255.255.255.0
AX(config-if:ve20)#ip nat outside
AX(config-if:ve20)#interface ve 31
AX(config-if:ve31)#ip address 100.64.1.3 255.255.255.0
AX(config-if:ve31)#ip nat inside
AX(config-if:ve31)#exit
```

Note: *Since VLANs are in use, the IP configuration and the IP NAT statements are associated with the virtual interfaces. If VLANs are not used, then place the IP NAT statements at the physical interface configuration level.*

8. Verify CGN configuration and operation using the following commands:

    - **show class-list**
    - **show ip nat interfaces**
    - **show session**
    - **show ip nat lsn full-cone-sessions**
    - **show ip nat lsn pool-statistics**
    - **show ip nat lsn user**
    - **show ip nat lsn statistics**

    Output examples for each command are shown below.

To show class-list configuration information:

```
AX(config)#show class-list
Name                             IP       Subnet   Location
vm_client_cgn01                  1        0        config
Total: 1
```

To show IP NAT interface information:

```
AX(config)#show ip nat interfaces
Total IP NAT Interfaces configured: 2
Interface       NAT Direction
-----------------------------
ve20            outside
ve31            inside
```

To show session information:

```
AX(config)#show session
Traffic Type                       Total
--------------------------------------------
TCP Established                    7
TCP Half Open                      0
UDP                                1030
Non TCP/UDP IP sessions            0
Other                              0
Reverse NAT TCP                    0
Reverse NAT UDP                    0
```

```
Curr Free Conn                     66877947
Conn Count                         1005042
Conn Freed                         1004005
TCP SYN Half Open                  0
Conn SMP Alloc                     0
Conn SMP Free                      0
Conn SMP Aged                      0
Conn Type 0 Available              133103616
Conn Type 1 Available              66877947
Conn Type 2 Available              33357819
Conn Type 3 Available              16637952
Conn SMP Type 0 Available          133103616
Conn SMP Type 1 Available          66551808
Conn SMP Type 2 Available          33275904
Conn SMP Type 3 Available          16646120

Prot Forward Source         Forward Dest          Reverse Source         Reverse Dest
Age   Hash Flags
-----------------------------------------------------------------------------------------
-----------------------
Udp  89.142.33.172:17838    192.0.2.33:14222   100.64.101.1:14222     89.142.33.172:17838
300   1    NF
Udp  87.1.181.162:39174     192.0.2.33:14111   100.64.100.1:14111     87.1.181.162:39174
240   1    NF
Udp  100.64.101.1:14222     81.88.222.83:59853     81.88.222.83:59853
192.0.2.33:14222   120   1    NF
Udp  81.235.197.227:24948   192.0.2.33:14222   100.64.101.1:14222
81.235.197.227:24948   240   1    NF
Udp  49.49.100.77:11009     192.0.2.33:14111   100.64.100.1:14111     49.49.100.77:11009
300   1    NF
```

To show information about full-cone sessions:

```
AX(config)#show ip nat lsn full-cone-sessions
LSN Full Cone Sessions:
Prot Inside Address         NAT Address            Conns  Pool             CPU Age
-------------------------------------------------------------------------------------
TCP  100.64.100.1:49293     192.0.2.33:49293   1      cgn-dynamic      4   -
UDP  100.64.100.1:14111     192.0.2.33:14111   415    cgn-dynamic      3   -
TCP  100.64.101.1:64527     192.0.2.33:64527   0      cgn-dynamic      3   0
TCP  100.64.101.1:64536     192.0.2.33:64536   0      cgn-dynamic      5   0
UDP  100.64.101.1:60399     192.0.2.33:60399   1      cgn-dynamic      4   -
```

To show CGN pool statistics:

```
AX(config)#show ip nat lsn pool-statistics
LSN Address Pool Statistics:
----------------------------
cgn-dynamic      Address           Users    ICMP     Freed    Total    UDP
Freed    Total    Rsvd     TCP      Freed    Total    Rsvd
--------------------------------------------------------------------------------
---------------------------------------------------
                 192.0.2.33   2        0        6        6        3        1215
1218     0        17       15466    15483
```

To show CGN client (end-user) information:

```
AX(config)#show ip nat lsn user top 2 all
LSN User-Quota Sessions:
Inside Address       NAT Address          ICMP   UDP    TCP    Session Pool
LID
----------------------------------------------------------------------------------------
100.64.101.1         192.0.2.33       0      2      10     590     cgn-dynamic      1
100.64.100.1         192.0.2.33       0      1      1      435     cgn-dynamic      1

AX#show ip nat lsn user inside-user 100.64.101.1
LSN User-Quota Sessions:
Inside Address       NAT Address          ICMP   UDP    TCP    Session Pool
LID
-----------------------------------------------------------------------------------------
---
100.64.101.1         192.0.2.33       0      2      11     805     cgn-dynamic      1
```

To show CGN statistics:

```
AX(config)#show ip nat lsn statistics
Traffic statistics for LSN:
---------------------------
Total TCP Ports Allocated                15488
Total TCP Ports Freed                    15476
Total UDP Ports Allocated                1218
Total UDP Ports Freed                    1215
Total ICMP Ports Allocated               6
Total ICMP Ports Freed                   6
Data Session Created                     996212
Data Session Freed                       994621
```

```
User-Quota Created                       4
User-Quota Freed                         2
User-Quota Creation Failed               0
TCP NAT Port Unavailable                 0
UDP NAT Port Unavailable                 0
ICMP NAT Port Unavailable                0
New User NAT Resource Unavailable        0
TCP User-Quota Exceeded                  260
UDP User-Quota Exceeded                  0
ICMP User-Quota Exceeded                 0
Extended User-Quota Matched              0
Extended User-Quota Exceeded             0
Data Session User-Quota Exceeded         0
TCP Full-cone Session Created            15488
TCP Full-cone Session Freed              15476
UDP Full-cone Session Created            1218
UDP Full-cone Session Freed              1215
Full-cone Session Creation Failed        0
Hairpin Session Created                  3
Self-Hairpinning Drop                    0
Endpoint-Independent Mapping Matched     347522
Endpoint-Independent Filtering Matched   631981
Endpoint-Dependent Filtering Drop        0
NAT Pool Mismatch Drop                   0
TCP Port Overloaded                      0
UDP Port Overloaded                      0
TCP Port Overloading Session Created     0
UDP Port Overloading Session Created     0
TCP Port Overloading Session Freed       0
UDP Port Overloading Session Freed       0
NAT IP TCP Max Ports Allocated           0
NAT IP UDP Max Ports Allocated           0
Full-cone Inbound Filtering Drop         0
No Class-List Match                      0
LSN LID Drop                             0
LSN LID Pass-through                     0
```

4.5 CONFIGURING FIXED-NAT

Fixed-NAT is a CGN feature that allocates NAT ports for each client from a predetermined (“fixed”) set of ports on the NAT address. Since each client using Fixed-NAT receives a deterministic set of ports, a client can be identified without any need for logging. Each individual client can be identified based solely on the NAT IP address and the port numbers within the client’s fixed allocation of ports. Fixed-NAT can be configured with a single command. To enable Fixed-NAT, use a command such as the one shown below to specify the inside address range, netmask, outside address range, netmask, ports per user, and HA group.

4.5.1 FIXED-NAT CONFIGURATION STEPS

To configure and verify Fixed-NAT, use the following steps.

1. Configure Fixed-NAT operation using the following command.

```
AX(config)#fixed-nat inside 100.64.101.1 100.64.101.1 netmask /32 nat
192.0.2.49 192.0.2.62 netmask /28 ports-per-user 1000 ha-group-id 1
```

Note: *The* ***port-per-user*** *command allows the operator to manually configure the port block allocation per inside address. If this command is not used, the software automatically calculates the number of ports for allocation based upon the number of inside and outside address ports that are available. See* Fixed-Nat Logging *for more information.*

2. Verify Fixed-NAT operation using the following commands:

- **show fixed-nat statistics**
- **show fixed-nat nat-address**
- **show fixed-nat inside-user**

Output examples for each command are shown below.

To show Fixed-NAT statistics:

```
AX(config)#show fixed-nat statistics
Fixed NAT Statistics:
----------------------------
Total TCP Ports Allocated                    5914
Total TCP Ports Freed                        5912
Total UDP Ports Allocated                    435067
Total UDP Ports Freed                        435032
Total ICMP Ports Allocated                   12
Total ICMP Ports Freed                       12
```

```
NAT44 Data Sessions Created                        526642
NAT44 Data Sessions Freed                          526605
NAT64 Data Sessions Created                        0
NAT64 Data Sessions Freed                          0
TCP NAT Port Unavailable                           0
UDP NAT Port Unavailable                           0
ICMP NAT Port Unavailable                          0
TCP User Quota Exceeded                            0
UDP User Quota Exceeded                            0
ICMP User Quota Exceeded                           0
Sessions User Quota Exceeded                       0
NAT44 TCP Full-Cone Created                        0
NAT44 TCP Full-Cone Freed                          0
NAT44 UDP Full-Cone Created                        1
NAT44 UDP Full-Cone Freed                          1
NAT44 UDP ALG Full-Cone Created                    0
NAT44 UDP ALG Full-Cone Freed                      0
NAT64 TCP Full-Cone Created                        0
NAT64 TCP Full-Cone Freed                          0
NAT64 UDP Full-Cone Created                        0
NAT64 UDP Full-Cone Freed                          0
NAT64 UDP ALG Full-Cone Created                    0
NAT64 UDP ALG Full-Cone Freed                      0
Full-Cone Session Creation Failed                  0
NAT44 Endpoint-Independent-Mapping Matched         272
NAT64 Endpoint-Independent-Mapping Matched         0
NAT44 Endpoint-Independent-Filtering Matched       90
NAT64 Endpoint-Independent-Filtering Matched       0
NAT44 Endpoint-Dependent Filtering Drop            0
NAT64 Endpoint-Dependent Filtering Drop            0
NAT44 Hairpin Session Created                      0
NAT64 Hairpin Session Created                      0
No Class-List Match                                0
Fixed NAT LID not Enabled                          0
Fixed NAT LID Standby Drop                         0
Self-Hairpinning Drop                              0
```

To show Fixed-NAT port-mapping information:

```
AX(config)#show fixed-nat nat-address 192.0.2.49 port-mapping
NAT IP Address: 192.0.2.49
Inside User: 100.64.101.1
 TCP:  1024 to 2023
 UDP:  1024 to 2023
 ICMP: 1024 to 2023
```

To show a specific Fixed-NAT port mapping by NAT address:

```
AX(config)#show fixed-nat nat-address 192.0.2.49 1566
Inside User: 100.64.101.1
```

To show a specific Fixed-NAT port mapping by client inside address:

```
AX(config)#show fixed-nat inside-user 100.64.101.1 port-mapping
NAT IP Address: 192.0.2.49
 TCP:  1024 to 2023
 UDP:  1024 to 2023
 ICMP: 1024 to 2023
```

To show user-quota usage:

```
AX(config)#show fixed-nat inside-user 100.64.101.1 quota-used
NAT IP Address: 192.0.2.49
Session Quota Used:  18
TCP Ports Used:      2
UDP Ports Used:      16
ICMP Resources Used: 0
```

Note: *All configuration options for Fixed-NAT, including EIM/EIF and ALG support, can be executed with **ip nat lsn** commands at the global configuration level.*

4.6 LOGGING CONFIGURATION

CGN traffic logs can be sent only to external log servers. If the AX device is configured to use a group of external log servers, it load balances the messages across the servers. Source-IP based hashing is used to select an external log server. This method ensures that traffic logs for a given source IP address always are directed to the same log server. Configuring the AX device for CGN traffic logs involves the following steps.

- Create a server configuration for each log server.
- Configure a service group and add the log servers to the group. The service group can contain a maximum of 32 members for external logging.
- Configure a logging template. Within the template, specify the service group and the types of events to log.
- Activate the template.

4.6.1 LOGGING CONFIGURATION STEPS

To configure and verify CGN external logging, use the following steps.

1. At the global configuration level, add a log server to the configuration. A name and IP address must be specified.

   ```
   AX(config)#slb server syslog1 100.64.100.1
   ```

   At the real server configuration level, specify the port and protocol for the syslog service. By default, these arguments are port “514” and protocol “UDP”. If a non-standard syslog port is required, the operator may modify the port number to match the logging environment.

   ```
   AX(config-real server)#port 514 udp
   AX(config-real server-node port)#exit
   AX(config-real server)#exit
   ```

2. At the global configuration level, create the service group and add the server to the group created in step 1. Specify the group name and protocol.

   ```
   AX(config)#slb service-group syslog udp
   ```

   Add the member to the group. Specify the server name given in step 1 and port number.

   ```
   AX(config-slb service group)#member syslog1:514
   AX(config-slb service group)#exit
   ```

3. Create the logging template and specify the syslog server group and the events to be logged. In this example, the service group name is "syslog" and both CGN events (log sessions and Fixed-NAT events) are logged. Alternatively, logging formats, RADIUS logging, source-port for syslog, and so on also can be modified at this configuration level. Please consult the Advanced CGN logging section and the *AX Series IPv4-to-IPv6 Transition Solutions Guide* for more information.

```
AX(config)#ip nat template logging LSN_LOG
AX(config-nat logging)#log sessions
AX(config-nat logging)#log fixed-nat-all
AX(config-nat logging)#log fixed-nat-user-ports
AX(config-nat logging)#service-group syslog
AX(config-nat logging)#exit
```

4. Activate the template by entering the following command at the global configuration level. Use the template name given in step 3.

```
AX(config)#ip nat lsn logging default-template LSN_LOG
```

Note: *The template will be applied to all IPv6 migration logging, including CGN, NAT64, and DS-Lite.*

5. View logging statistics:

```
AX#show ip nat logging statistics
NAT Logging Statistics:
----------------------------
TCP Session Created                      24934
TCP Session Deleted                      24925
TCP Port Allocated                       15657
TCP Port Freed                           15632
TCP Port Batch Allocated                 0
TCP Port Batch Freed                     0
UDP Session Created                      978222
UDP Session Deleted                      976743
UDP Port Allocated                       1235
UDP Port Freed                           1217
UDP Port Batch Allocated                 0
UDP Port Batch Freed                     0
ICMP Session Created                     6
ICMP Session Deleted                     6
ICMP Resource Allocated                  6
ICMP Resource Freed                      6
ICMPV6 Session Created                   0
ICMPV6 Session Deleted                   0
```

```
ICMPV6 Resource Allocated                0
ICMPV6 Resource Freed                    0
GRE Session Created                      0
GRE Session Deleted                      0
GRE Resource Allocated                   0
GRE Resource Freed                       0
ESP Session Created                      0
ESP Session Deleted                      0
ESP Resource Allocated                   0
ESP Resource Freed                       0
Fixed NAT Inside User Port Mapping       0
Fixed NAT Disabled Config Logged         1
Fixed NAT Disabled Config Logs Sent      1
Log Packets Sent                         2
Log Packets Dropped                      0
Tcp Connection Established               0
Tcp Connection Lost                      0
TCP Port Overloading Allocated           0
TCP Port Overloading Freed               0
UDP Port Overloading Allocated           0
UDP Port Overloading Freed               0
```

5 ADVANCED CONFIGURATION OPTIONS

This section presents the following advanced configuration options:

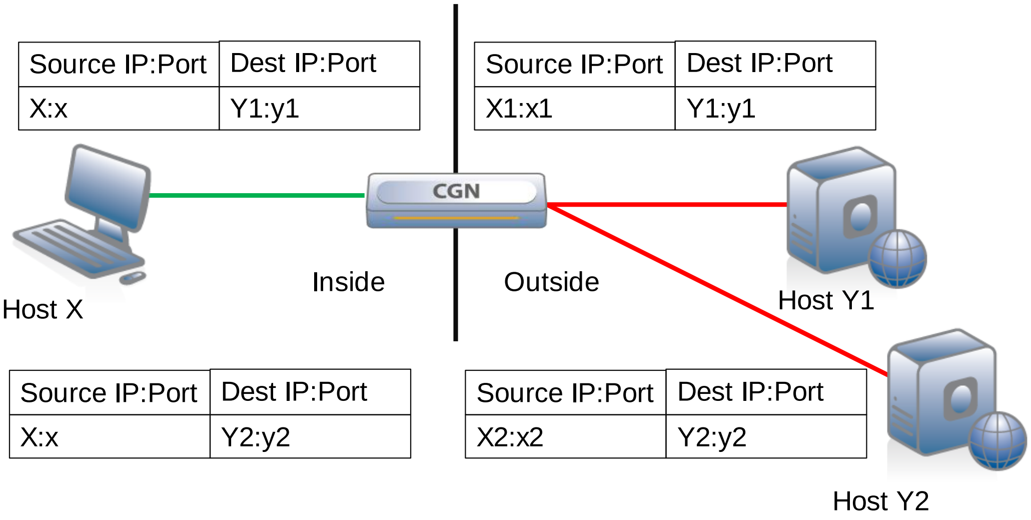
- EIM/EIF
- Static mapping
- Override options
- NAT address selection method
- Hairpinning
- User quotas
- Application Layer Gateways (ALGs)
- Protocol port overload

- CGN timeouts
- System resource allocation
- Advanced CGN Logging

5.1 ENDPOINT-INDEPENDENT MAPPING / ENDPOINT-INDEPENDENT FILTERING

Endpoint-independent Mapping (EIM) and Endpoint-independent Filtering (EIF) provide crucial behavioral characteristics for CGN and should be considered mandatory options for most applications. EIM provides a stable, long-term binding where internal hosts may connect by utilizing the same NAT binding for multiple external hosts (as long as the internal port does not change). However, if the internal port changes, CGN is free to create a new binding and thus a new port is assigned.

In Figure 3, EIM behavior is illustrated. Host X initiates a conversation with Host Y1 and is assigned an address/port from the NAT pool of X1:x1. Then, the application initiates the same connection with host Y2, using the same source port. This is typical for peer-to-peer applications and some Internet messenger protocols. Since the internal port of Host X remains unchanged, the original NAT binding of X1:x1 is used for traffic to Host Y2.



Source IP:Port	Dest IP:Port
X:x	Y1:y1

Source IP:Port	Dest IP:Port
X1:x1	Y1:y1

Source IP:Port	Dest IP:Port
X:x	Y2:y2

Source IP:Port	Dest IP:Port
X2:x2	Y2:y2

EIM implies X1:x1 = X2:x2 for all Y:y (Y1:y1 and Y2:y2)

Figure 3: EIM model

- EIM provides a stable, long-term binding that an internal host may use for connection to external servers.
- EIF is closely related to EIM, and controls which external servers may access a host using an established binding.

Figure 4 shows that a NAT binding has been created for the traffic passing between Host A and Host B using NAT IP address X and port 9001. EIF (full-cone behavior) allows for any port on Host B or any port on Host C to use the original NAT binding. In essence, the external host's address/port is irrelevant and is treated as a wildcard. Traffic will pass from any external address/port, as long as it is addressed to the NAT address:port X:9001.

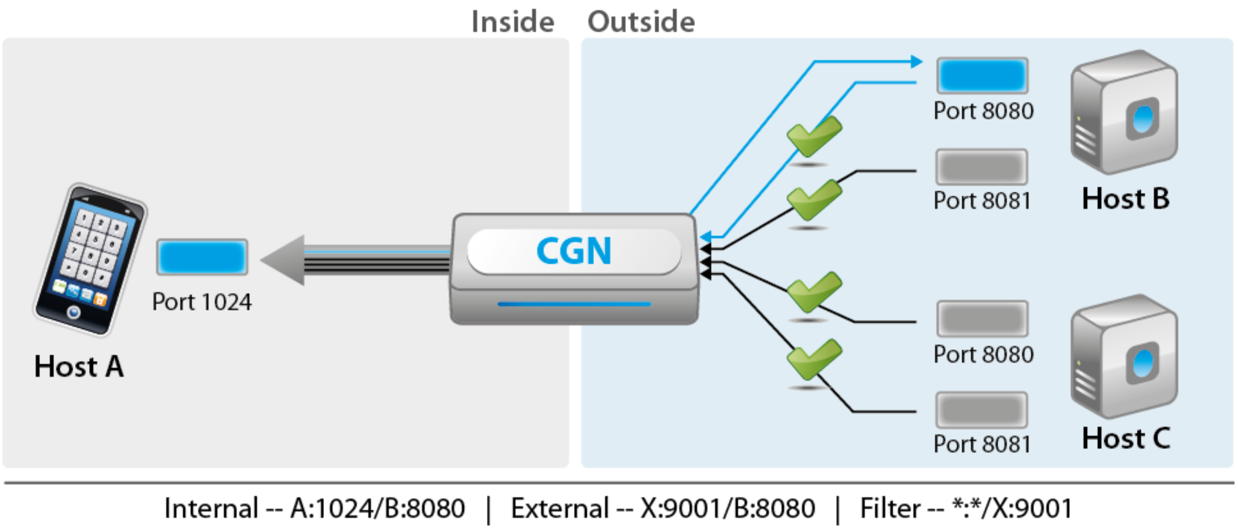


Figure 4: EIF model

By default, full-cone support (EIM/EIF) is disabled on well-known TCP and UDP ports (1-1023), and is enabled on ephemeral ports (1024-65535).

EIM/EIF can be configured together or individually, for any ports.

5.1.1 ENABLING OR DISABLING EIM/EIF

To enable EIM/EIF simultaneously, use the following command:

```
AX(config)#ip nat lsn full-cone enable
```

Likewise, to disable them simultaneously, use the following command:

```
AX(config)#ip nat lsn full-cone disable
```

To enable or disable EIM only, use the following commands:

```
AX(config)#ip nat lsn endpoint-independent-filtering enable
AX(config)#ip nat lsn endpoint-independent-filtering disable
```

Likewise, to enable or disable EIF only, use the following commands:

```
AX(config)#ip nat lsn endpoint-independent-mapping enable
AX(config)#ip nat lsn endpoint-independent-mapping disable
```

Note: *Address pooling should not be confused with EIM/EIF behavior. Address pooling attempts to use the same external NAT IP address for all flows from a particular internal client. Unlike EIM/EIF, address pooling is not concerned with ports but instead is concerned only with a consistent IP address mapping between the internal client and the external NAT address. This solves the issue of multiple flows from a client (sent over NAT) to a server that is expecting the flows to be from the same source address. For example, SIP clients often use multiple source ports for RTCP and RTP, and if these flows do not originate from the same external source address (the address assigned from the NAT pool), the destination may drop the traffic.*

5.2 STATIC MAPPING

To ensure that a service on the inside is available at a fixed outside IP/port pair, a static mapping can be configured. Static mapping is supported by all AX Series CGN devices.

To enable static mappings, use the following command at the global configuration level:

```
ip nat lsn port-reservation inside priv-ipaddr start-priv-portnum end-priv-portnum nat
public-ipaddr start-public-portnum end-public-portnum
```

For example:

```
AX(config)#ip nat lsn port-reservation inside 100.64.100.1 1024 2000 nat 192.0.2.32
1024 2000
```

5.3 OVERRIDE ACTIONS FOR CLASS-LIST MATCHES

By default, when traffic matches a class list, the source address is subject to NAT. The override function allows for alternative actions, such as passing through or dropping traffic that matches the class list.

To drop all traffic matching a class list, apply the following command at the LSN-LID configuration level:

```
override drop
```

For example:

```
AX(config)#lsn-lid 1
AX(config-lsn lid)#override drop
```

Likewise, to pass through and route (without NAT) all traffic that matches a class list, use the following command:

```
override pass-through
```

For example:

```
AX(config)#lsn-lid 1
AX(config-lsn lid)#override pass-through
```

Note: *The AX Series also supports an enhanced feature (for override options), which utilizes source and destination matching. Please refer to the "AX Series IPv4-to-IPv6 Transition Solutions Guide" for more information.*

5.4 NAT IP ADDRESS SELECTION

By default, the AX Series randomly chooses the NAT IP address from the configured pool of addresses. To provide configuration flexibility for efficient use of public addresses, the following additional IP address selection methods are supported:

- Random – random (long-run uniformly distributed)
- Round-robin – round-robin
- Least-used-strict – fewest NAT ports used
- Least-UDP-used-strict – fewest UDP NAT ports used
- Least-TCP-used-strict – fewest TCP NAT ports used
- Least-reserved-strict – fewest NAT ports reserved
- Least-UDP-reserved-strict – fewest UDP NAT ports reserved
- Least-TCP-reserved-strict – fewest TCP NAT ports reserved
- Least-users-strict – fewest users

For example, to configure the round-robin address selection method, use the following command at the global configuration level:

```
AX(config)#ip nat lsn ip-selection round-robin
```

5.5 HAIRPINNING

Hairpinning is enabled by default and can be configured to prevent self-hairpinning, meaning that an inside client's traffic cannot be rerouted to itself. There are three filtering options that can be used to change the behavior: Self-IP, Self-IP-port, and none (default).

- Self-IP filtering drops traffic from a client to its own NAT address regardless of which port is in use. This option applies to both UDP and TCP traffic.
- Self-IP-port filtering drops traffic only if the destination is the client's own public IP address, and the source IP address and protocol port are the address and port used in the client's NAT mapping. This option is useful in cases where double NAT is used. In this case, more than one client might be behind a single NAT IP address and hairpinning traffic between the two clients is legitimate, even though from the CGN perspective the client's traffic is hairpinned back to itself.

The default behavior is NONE and is characterized as follows:

- UDP traffic – 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 that was used in the mapping for the client.
- TCP traffic – Self-IP-port hairpin filtering is used for TCP traffic.

To configure hairpinning filter options, use the following command at the global configuration level:

```
ip nat lsn hairpinning {filter-none | filter-self-ip | filter-self-ip-port}
```

For example, to configure Self-IP filtering for hairpinning, use the following command at the global configuration level:

```
AX(config)#ip nat lsn hairpinning filter-self-ip
```

5.6 USER QUOTAS

CGN user quotas limit the number of NAT port mappings allowed for individual internal IP addresses. For example, each inside IP address can be limited to a maximum of 100 TCP NAT ports. Once a client reaches the quota, the client is not allowed to open additional TCP sessions. User quotas can be configured for TCP, UDP, and ICMP protocols on a global basis or on a per-LID assignment basis.

When an inside client initiates a session, the entire quota value is allocated to that client. This limits the number of inside clients that can be supported per NAT IP address. To alleviate this issue, the operator may choose to reserve a subset of the total quota, thus freeing the remainder of the ports to be used by another client. This method allows for more efficient use of NAT IP address resources. Please see the *AX Series IPv4-to-IPv6 Transition Solutions Guide* for more detail.

Once a client reaches its quota for a particular protocol, no new translations are allowed. To ensure that ports are available for critical services, extended quotas can be specified for source protocol/port.

To configure a user quota, issue the following command at the LSN-LID configuration level:

```
user-quota protocol quota-num [reserve reserve-num]
```

For example:

```
AX(config)#lsn-lid 1
AX(config-lsn lid)#user-quota tcp 1000 reserve 100
```

In this example, inside client TCP traffic is limited to 1000 ports per client. One hundred ports are immediately reserved while the remaining 900 ports are free to be used by other clients. Optionally, configure extended quota for critical services:

```
AX(config-lsn lid)#extended-user-quota tcp service-port 25 sessions 5
```

This command allows an additional 5 ports to be made available to email services once the quota is reached.

Due to the nature of EIM/EIF, it is possible for inside or outside devices to set up more sessions than the allotted quota. The **session** option limits the total number of sessions, including full-cone sessions. Use the following command to set a total session limit:

```
user-quota session num
```

For example:

```
AX(config)#lsn-lid 1
AX(config-lsn lid)#user-quota session 5000
```

By default, if a client exceeds the user quota, an ICMP destination unreachable message is sent to the source. To disable this behavior, use the following command:

```
ip nat lsn icmp send-on-user-quota-exceeded disable
```

5.7 THE APPLICATION LAYER GATEWAY

An Application Layer Gateway (ALG) is a feature that changes the payload in a packet to ensure that the protocol will continue to work over NAT. Usually, the IP addresses and protocol port numbers are communicated in the payload of a packet, as part of the application protocol. However, if the address information is translated by the NAT gateway, this will inherently cause problems due to the mismatching addresses. The AX Series provides 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)

FTP is supported by default. To enable additional ALG support for LSN and Fixed-NAT applications, use the following command:

```
ip nat lsn alg {esp | ftp | pptp | rtsp | sip | tftp} {enable | disable}
```

Note: *ESP ALG currently is not supported for Fixed-NAT deployments. It is assumed that most clients will take advantage of IPsec NAT-T for IPsec support.*

5.8 PROTOCOL PORT OVERLOADING

When public IP addresses are scarce and the number of inside clients exceeds the total number of available NAT ports, protocol port overloading provides an efficient port sharing mechanism. Protocol port overloading enables the AX device to use the same NAT IP port for more than one user if the destinations are unique. This behavior is illustrated in Figure 5, where clients A:a and B:b are sending traffic to Server X and Y respectively. In this case, the NAT IP address and port can be used for both clients, A and B.

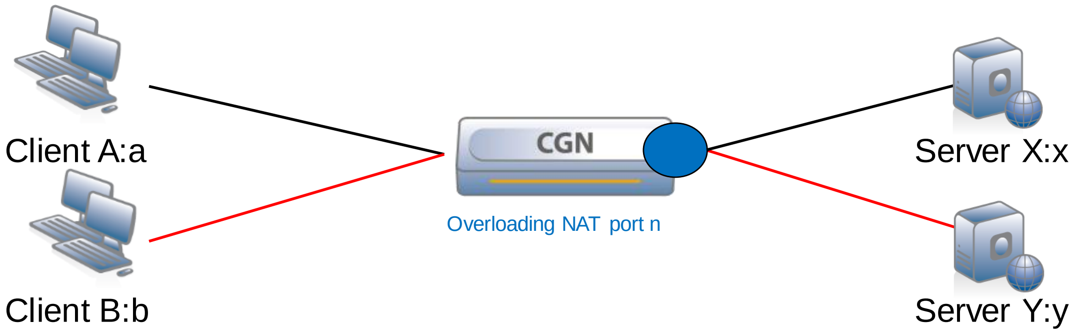


Figure 5: Port overloading

Port overloading works well in environments where the service provider has few public IP addresses for NAT, the majority of the traffic is client-server, and there are no peer-to-peer applications. Port overloading can be configured for all destination ports, well-known ports only, UDP/TCP, or for a specific range of ports.

5.8.1 EXAMPLES

To enable protocol port overloading for all destination ports:

```
ip nat lsn port-overloading enable
```

To enable for well-known destination ports only:

```
ip nat lsn port-overloading enable well-known
```

To enable for only TCP destination port 80:

```
ip nat lsn port-overloading tcp enable 80
```

5.8.2 PORT OVERLOADING CONFIGURATION OPTIONS

The default behavior is to overload a port only when the IP address is unique. To allow port overloading behavior when more than one session is directed to the same external server, enable the **destination-address-and-port** option:

```
ip nat lsn port-overloading unique destination-address-and-port
```

By default, a port can be overloaded to create multiple mappings only for the same client. To allow an overloaded port to be used by more than one client, use the following command at the global configuration level:

```
ip nat lsn port-overloading allow-different-user
```

5.8.3 VERIFYING OPERATION

To verify operation, use the following command:

show ip nat lsn statistics

For example:

```
AX(config)#show ip nat lsn statistics
Traffic statistics for LSN:
---------------------------
Total TCP Ports Allocated                23133
Total TCP Ports Freed                    23101
Total UDP Ports Allocated                1395
Total UDP Ports Freed                    1392
Total ICMP Ports Allocated               6
Total ICMP Ports Freed                   6
Data Session Created                     1353616
Data Session Freed                       1352772
User-Quota Created                       6
User-Quota Freed                         4
User-Quota Creation Failed               0
TCP NAT Port Unavailable                 0
UDP NAT Port Unavailable                 0
ICMP NAT Port Unavailable                0
New User NAT Resource Unavailable        0
TCP User-Quota Exceeded                  260
UDP User-Quota Exceeded                  0
ICMP User-Quota Exceeded                 0
Extended User-Quota Matched              0
Extended User-Quota Exceeded             0
Data Session User-Quota Exceeded         0
TCP Full-cone Session Created            23133
TCP Full-cone Session Freed              23101
UDP Full-cone Session Created            1392
UDP Full-cone Session Freed              1389
Full-cone Session Creation Failed        0
Hairpin Session Created                  4
Self-Hairpinning Drop                    0
Endpoint-Independent Mapping Matched     465289
Endpoint-Independent Filtering Matched   863797
```

```
Endpoint-Dependent Filtering Drop        0
NAT Pool Mismatch Drop                   0
TCP Port Overloaded                      0
UDP Port Overloaded                      1
TCP Port Overloading Session Created     0
UDP Port Overloading Session Created     10
TCP Port Overloading Session Freed       0
UDP Port Overloading Session Freed       9
NAT IP TCP Max Ports Allocated           0
NAT IP UDP Max Ports Allocated           0
Full-cone Inbound Filtering Drop         0
No Class-List Match                      0
LSN LID Drop                             0
LSN LID Pass-through                     0
```

5.8.4 CONSIDERATIONS

- Port overloading is not compatible with EIM/EIF. If port overloading is configured, EIM/EIF will be disabled.
- Port overloading enable/disable requires a reload of the AX device in order to take effect. Use the following command to determine the configuration state:

  ```
  show ip nat lsn port-overloading config
  ```

- The AX device will only overload ports when either the user quota is exceeded or there are no more free ports.
- Port Batching is not compatible with the **allow-different-user** option.

5.9 CGN TIMEOUTS

The AX Series allows for NAT timer reconfiguration to ensure proper application operation for varying network environments. This includes NAT session timeouts, STUN timeouts, and SYN timeout.

5.9.1 NAT SESSION TIMEOUTS

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

- For a TCP session, the data session is removed when the AX device observes the FIN or RST messages exchanged by the two endpoints of the session. If the AX device 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.

NAT session aging is individually configurable for TCP, UDP, and ICMP, using the **ip nat translation** command.

- **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 default is fast (2 seconds).
- **service-timeout** – Custom service timeout for any individual port.

Note: *DNS defaults to a timeout of fast (3 seconds in this case).*

For example, to configure TCP port 80 for a timeout of 120 seconds, use the following command:

```
AX(config)#ip nat translation service-timeout tcp 80 to 80 120
```

To view current NAT session aging timeout status, use the following command:

```
AX(config)#show ip nat timeouts
NAT Timeout values in seconds:
SYN     TCP     UDP     ICMP
------------------------
60      300     300     fast
Service 53/udp has fast-aging configured
```

5.9.2 STUN TIMEOUT

The STUN timeout specifies how long a NAT mapping for a full-cone session is maintained after the data session ends. The default is 2 minutes.

To configure a STUN timeout of 4 minutes for all ports, issue the following command:

```
ip nat lsn stun-timeout seconds
```

For example:

```
AX(config)#ip nat lsn stun-timeout 4
```

5.9.3 SYN IDLE TIMEOUT

CGN supports a SYN idle timeout to control “half-open” situations and to provide protection against SYN flood attacks. If a TCP session is not established within the configured time period, the AX device drops the session. The SYN idle timeout can be set from 2-7 seconds, and is 4 seconds by default.

For example, to change the CGN timeout to 7 seconds, use the following command at the global configuration level:

```
AX(config)#ip nat lsn syn-timeout 7
```

5.10 SYSTEM RESOURCE ALLOCATION

The AX Series allows for configuration of limiting system resources, including the maximum number of sessions and allocated NAT pool addresses. The maximum number allowed for these resources varies for each AX model. To display the maximum for your AX device, use the following command:

```
AX(config)#show system resource-usage
Resource                         Current    Default    Minimum    Maximum
--------------------------------------------------------------------------
l4-session-count                 67108864   33554432   8388608    67108864
nat-pool-addr-count              4000       500        500        4000
real-server-count                1024       1024       512        8192
real-port-count                  2048       2048       512        16384
service-group-count              512        512        512        8192
virtual-port-count               1024       1024       256        8192
virtual-server-count             512        512        512        4096
http-template-count              256        256        32         4096
proxy-template-count             256        256        32         4096
conn-reuse-template-count        256        256        32         4096
fast-tcp-template-count          256        256        32         4096
fast-udp-template-count          256        256        32         4096
client-ssl-template-count        256        256        32         8192
server-ssl-template-count        256        256        32         8192
stream-template-count            256        256        32         4096
persist-cookie-template-count    256        256        32         4096
persist-srcip-template-count     256        256        32         4096
class-list-ipv6-addr-count       2048000    2048000    2048000    4096000
```

In this example the Layer 4 session allocation (l4-session-count) and NAT pool address limit (nat-pool-addr-count) have been configured for the maximum values.

To adjust the resource allocations, use the following commands:

```
system resource-usage nat-pool-addr-count num
system resource-usage l4-session-count num
```

For example:

```
AX(config)#system resource-usage nat-pool-addr-count 4000
AX(config)#system resource-usage l4-session-count 67108864
```

Note: *The total number of L4-sessions include both full-cone and user-quota sessions.* ***A reboot of the AX device is required for changes to system resource allocations.***

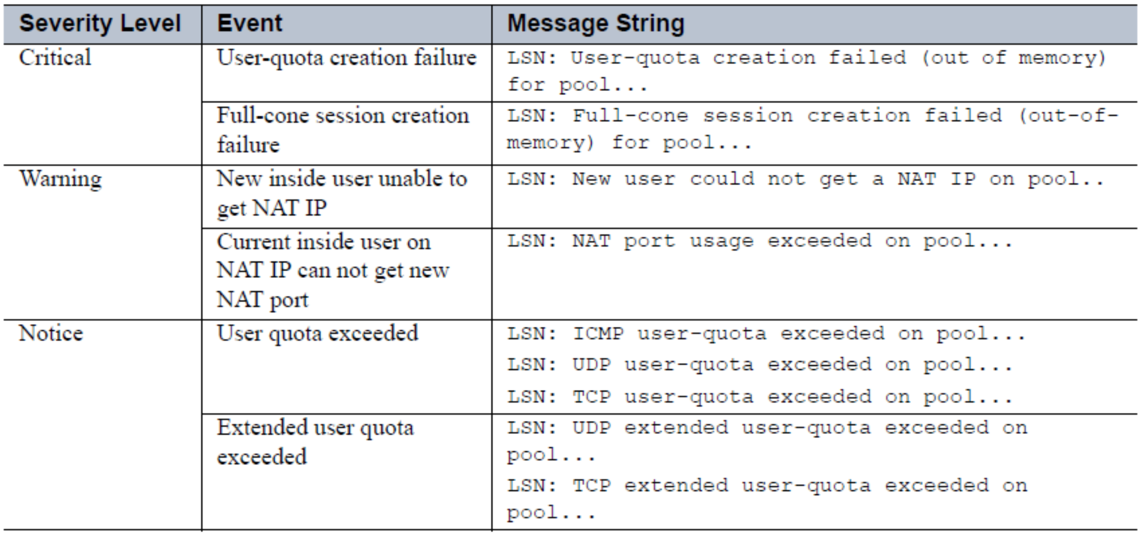
5.11 ADVANCED CGN LOGGING

CGN logging is a crucial functionality required by ISPs and carriers, who need to be able to determine the IP addresses and ports of their users at any given time. Generally, this type of record keeping is government mandated. There are multiple approaches for dealing with logging demands. Some customers require extensive logging, while other customers just need to be able to track a given connection at a given time back to a certain subscriber. The AX Series offer solutions for any of these scenarios.

5.11.1 CGN OPERATIONAL LOGGING

The AX Series supports both operational logging and CGN traffic logging. Operational logging utilizes the standard AX logging mechanism and can be written to the local logging buffer or target locations. Because of the volume of log messages generated from CGN, traffic logging is supported only to external servers.

CGN supports operational logging for resource failures. The following events are supported:



Severity Level	Event	Message String
Critical	User-quota creation failure	`LSN: User-quota creation failed (out of memory) for pool...`
	Full-cone session creation failure	`LSN: Full-cone session creation failed (out-of-memory) for pool...`
Warning	New inside user unable to get NAT IP	`LSN: New user could not get a NAT IP on pool..`
	Current inside user on NAT IP can not get new NAT port	`LSN: NAT port usage exceeded on pool...`

Notice	User quota exceeded	`LSN: ICMP user-quota exceeded on pool...`
		`LSN: UDP user-quota exceeded on pool...`
		`LSN: TCP user-quota exceeded on pool...`

	Extended user quota exceeded	`LSN: UDP extended user-quota exceeded on pool...`
		`LSN: TCP extended user-quota exceeded on pool...`

Figure 6: CGN operational logs (from 2.6.6 AX Series IPv4-to-IPv6 Transition Solutions Guide)

Basic operational logging can be enabled with the following command at the global configuration level:

```
logging buffered debugging
```

For details regarding logging targets and levels, see the *AX Series System Configuration and Administration Guide*.

5.11.2 CGN TRAFFIC LOGGING

Traffic logging includes all CGN session and NAT port mapping logs, and is supported only for external log servers. Currently, up to 32 log servers are supported. If multiple log servers are configured, the AX device load balances messages to all servers by utilizing source-IP based hashing. This ensures that traffic logs for a particular source IP address always are directed to the same server.

Configuration for traffic logging is covered in the Logging Configuration section of this document. For reference, here is an excerpt from an AX configuration file that includes the logging configuration shown in that section:

```
slb server syslog1 100.64.100.1
   port 514  udp
slb service-group syslog udp
    member syslog1:514
ip nat template logging LSN_LOG
 log sessions
 log fixed-nat-all
 log fixed-nat-user-ports
 service-group syslog
ip nat lsn logging default-template LSN_LOG
```

In this example, the syslog server is defined as 100.64.100.1 and is included as a member in the group “syslog”. In the template, dynamic sessions, Fixed-NAT sessions, and user ports are configured for logging. For more detail, please see the AX Series IPv4-to-IPv6 Transition Solutions Guide section.

5.11.3 LOG FILE SIZE REDUCTION

ASCII-formatted log files can lead to massive amounts of data, requiring massive storage requirements on the order of several terabytes of data per day. To alleviate these issues, A10 Networks offers log file size reduction techniques, including compact and binary logging.

By default, all traffic logging is presented in ASCII format. When a session is created or deleted, up to four log entries can be recorded. These include NAT Port Mapping creation/freed and session logs for creation/deletion for TCP/UDP/ICMP sessions. For example:

Example port mapping creation log:

```
Tstamp AX_hostname NAT-UDP-C: inside_ip:inside_port<-->nat_ip:nat_port to
dest_ip:dest_port
```

Example session creation log:

```
Tstamp AX_hostname NAT-TCP-C: fwd_src_ip:fwd_src_port<->fwd_dest_ip:fwd_dest_port,
rev_src_ip:rev_src_port<-->rev_dest_ip:rev_dest_port
```

For scaling purposes, assume the ASCII-formatted log entries for IPv4 are around 150 bytes, and assume the ASCII-formatted log entries for IPv6 are around 200 bytes.

5.11.4 COMPACT LOGGING FORMAT

Compact logging format reduces log size by using short operational codes for the event and protocol names, and hexadecimal format for the IPv4 addresses. IPv6 addresses continue to be shown in their original hexadecimal format. Compact logging results in an average of 33 percent reduction in log size.

Example ASCII log:

```
[timestamp] [hostname] NAT-UDP-C: 100.100.100.100:10000 -> 150.150.150.150: 10000
```

Example compact log:

```
[timestamp] [hostname] UC: 64646464:2710->96969696:2710
```

5.11.5 BINARY LOGGING FORMAT

Binary logging uses a unique binary format to efficiently reduce the size of log messages for IPv4 and IPv6.

Example:

To configure traffic logging using binary format, follow the configuration steps below:

1. Enter the configuration level for the logging template:

```
AX(config)#ip nat template logging LSN_LOG
```

2. Enable binary format:

```
AX(config-nat logging)#format binary
```

Note: *The **binary** option is given as an example. Valid options include **compact** (Hex logging), **default** (ASCII), and **rfc5424**.*

5.11.6 LOG VOLUME REDUCTION

The log message format can be changed to drastically reduce the size of the log file messages, as demonstrated in the previous section. However, while reducing the size of the individual log messages, log formats cannot reduce the number of log messages that are generated. In addition to log file reduction, the AX Series also supports log volume reduction through Port Batching and Fixed-NAT logging, which are discussed below.

5.11.7 PORT BATCHING

Port Batching reduces the amount of data created by the AX device's logging features by allocating a set of multiple ports to the client during session initiation, then generating only a single log message for the batch of ports. Batch sizes up to 1024 ports are supported.

Port Batching example:

```
Jan 23 13:27:35 AX5200-11 NAT-UDP-B: 30.30.30.11 -> 162.168.20.220:16251,16, 23
```

The following information is encoded in binary format:

- B – binding established
- 16251 – base port for this allocation

- 16 – batch size - number of ports assigned in this allocation
- 23 – step size - port increment i.e. ports assigned are 16521, 16544, 16567…… etc.

To enable and verify Port Batching:

Port Batching is disabled by default. To enable Port Batching with a batch size of 1024, use the following command at the global configuration level:

```
ip nat lsn port-batching size num
```

To view the status of Port Batching, use the following command:

```
AX(config)#show ip nat logging statistics
NAT Logging Statistics:
----------------------------
TCP Session Created                      83740
TCP Session Deleted                      83731
TCP Port Allocated                       53995
TCP Port Freed                           53965
TCP Port Batch Allocated                 233
TCP Port Batch Freed                     200
UDP Session Created                      2613915
UDP Session Deleted                      2611997
UDP Port Allocated                       2034
UDP Port Freed                           2026
UDP Port Batch Allocated                 111
UDP Port Batch Freed                     100
```

5.11.8 FIXED-NAT ADVANCED CONFIGURATION

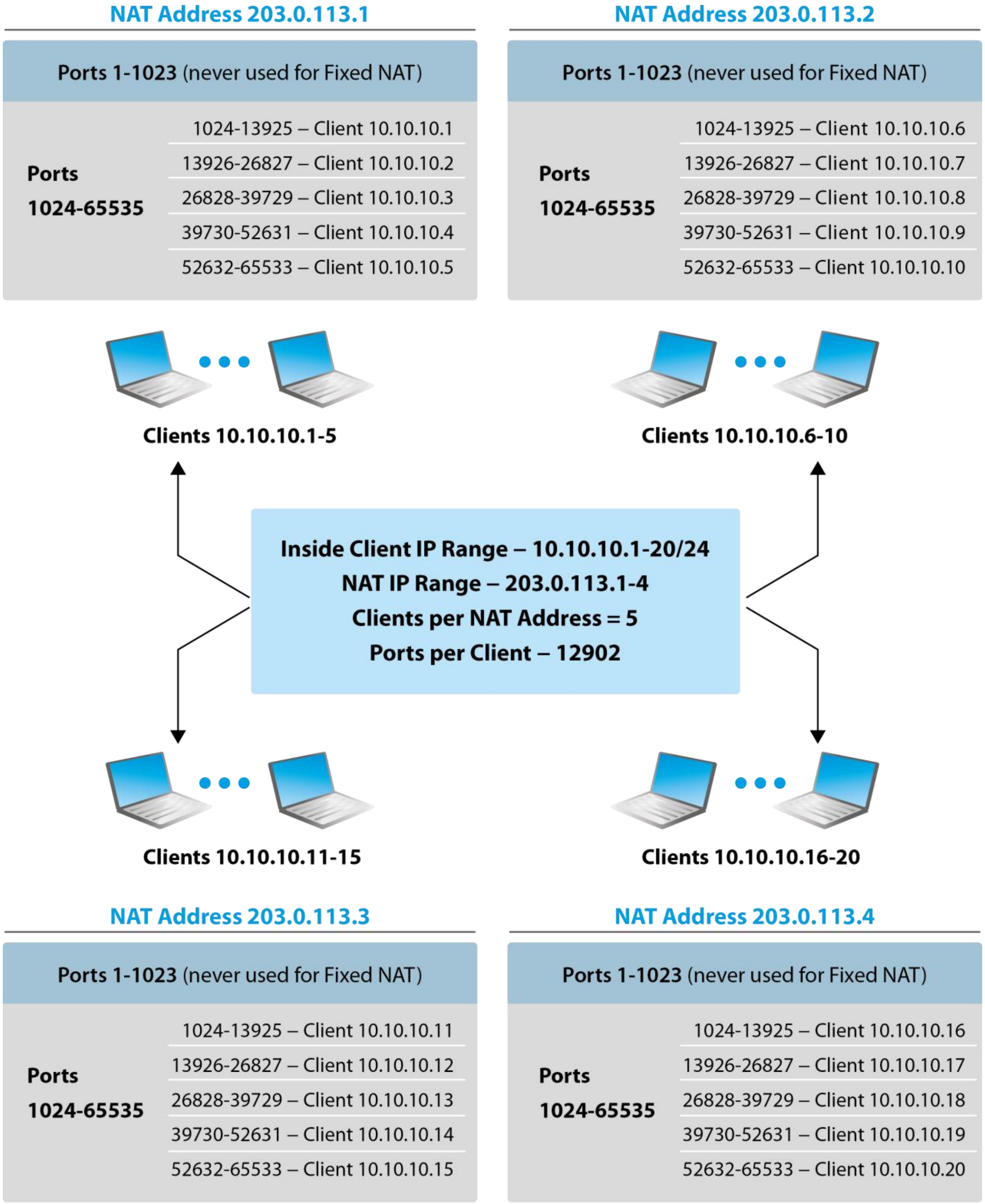
Fixed-NAT is a log optimization feature that allocates NAT ports for each client from a predetermined (“fixed”) set of ports on the NAT address. Since each client now receives a deterministic set of ports, a client can be identified without any need for logging. Each individual client can be identified based solely on the NAT IP address and the port numbers within the client’s fixed allocation of ports.

The implementation supports both manual and automatic port assignments. Manual port assignments are designated in blocks during configuration. For example, each inside address is assigned 1000 ports, thus resulting in the first inside IP address being associated with ports 1024-2023, and the second inside IP address with 2024-3023, and so on. Automatic configuration uses a simple algorithm to determine the block of ports for assignment. First, the number of inside clients is divided by the number of available outside NAT addresses. The result then is divided by the available ports per NAT address (default 64512). This can be expressed by the following formula:

```
P = 64512/RoundUp{IP(i)/IP(o)}
```

Where **IP(o)** is the number of available outside IP addresses, **IP(i)** is the number of inside IP addresses, and **P** is the number or ports available per **IP(i)**.

In the example illustrated in Figure 1**Error! Reference source not found.**, there are 20 inside clients that require mapping to 4 outside NAT addresses. Applying the formula above results in (64512)/(20/4) = 12902. The CGN device assigns the first range of ports on the beginning NAT address to the beginning client IP address. The next range of ports is assigned to the next client, and so on, until the NAT address does not have enough ports to add another client. In this case, the first range of ports on the next NAT address is used for the next inside client, and so on.



NAT Address 203.0.113.1

Ports 1-1023 (never used for Fixed NAT)	
Ports 1024-65535	1024-13925 – Client 10.10.10.1
	13926-26827 – Client 10.10.10.2
	26828-39729 – Client 10.10.10.3
	39730-52631 – Client 10.10.10.4
	52632-65533 – Client 10.10.10.5

NAT Address 203.0.113.2

Ports 1-1023 (never used for Fixed NAT)	
Ports 1024-65535	1024-13925 – Client 10.10.10.6
	13926-26827 – Client 10.10.10.7
	26828-39729 – Client 10.10.10.8
	39730-52631 – Client 10.10.10.9
	52632-65533 – Client 10.10.10.10

Inside Client IP Range – 10.10.10.1-20/24
NAT IP Range – 203.0.113.1-4
Clients per NAT Address = 5
Ports per Client – 12902

NAT Address 203.0.113.3

Ports 1-1023 (never used for Fixed NAT)	
Ports 1024-65535	1024-13925 – Client 10.10.10.11
	13926-26827 – Client 10.10.10.12
	26828-39729 – Client 10.10.10.13
	39730-52631 – Client 10.10.10.14
	52632-65533 – Client 10.10.10.15

NAT Address 203.0.113.4

Ports 1-1023 (never used for Fixed NAT)	
Ports 1024-65535	1024-13925 – Client 10.10.10.16
	13926-26827 – Client 10.10.10.17
	26828-39729 – Client 10.10.10.18
	39730-52631 – Client 10.10.10.19
	52632-65533 – Client 10.10.10.20

Figure 7: Port allocation model when Fixed-NAT is used

5.11.9 FIXED-NAT LOGGING

Fixed-NAT logging is supported to achieve compliance with either legal or company policy. Using Fixed-NAT significantly reduces the log volume for CGN deployments. When a client initiates its first session, a single log file is sent that captures all of the ports assigned to that individual inside address. No other logging activity occurs during the lifetime of the session. Also, if configuration changes occur that modify any attribute of Fixed-NAT, a Fixed-NAT-Disable log entry is sent to ensure that any changes that are made to assigned IP addresses and port allocations are captured.

Fixed-NAT user ports logging example:

```
FIXED-NAT-PORTS 10.10.10.172->192.168.9.173:3000-4000
```

Fixed NAT disable logging example:

```
FIXED-NAT-DISABLE 10.10.10.172->192.168.9.173
```

Note: *These examples illustrate the A10 ASCII format for log messages. However, Fixed-NAT also can take advantage of both the Compact and Binary logging formats discussed earlier, thus reducing log messages to the smallest size possible.*

The AX Series supports the following logging options for Fixed-NAT:

- Connection logging – To include session and port-mapping logging, use the following command at the configuration level for the logging template:

  ```
  log fixed-nat-all
  ```

- Port-map logging – To include port mapping information for private addresses, use the following command at the configuration level for the logging template:

  ```
  log fixed-nat-user-ports
  ```

5.11.10 SYSLOG (RFC 5424)

The AX Series supports the Syslog protocol (RFC 5424) for storing log events, including port mappings, port batching, Fixed-NAT enable/disable, and session creation or session deletion activity. Syslog provides a structured format for easier parsing, as well as more verbose information than standard logging, and full customization of syslog messages.

To configure traffic logging format based on RFC 5424, issue the following command at the configuration level for the logging template:

```
format rfc5424
```

The AX Series supports both the default timestamp and the RFC custom header alternate timestamp. The formats are as follows:

- Default – “2012-03-12T12:30:12-07:00”
- Alternate – “2012 Mar 12 12:30:12”

To configure the RFC custom header alternate timestamp, use the following command at the configuration level for the logging template:

```
rfc header use-alternate-timestamp
```

The AX Series supports full customization of message strings. Both text and fields can be added, modified, or deleted. The syntax structure includes the CLI command **rfc-custom message** *feature event*. To view the available feature event keywords, execute the following command:

```
AX#show ip nat logging keywords ?
  6rd-nat64           6rd-NAT64
  ds-lite             DS-Lite
  lsn                 LSN
  nat64               NAT64
  session-created     Session created
  session-deleted     Session deleted
```

To view specific keywords available for CGN, use the **lsn** option:

```
AX#show ip nat logging keywords lsn ?
  fixed-nat-allocated   Fixed-NAT allocated
  fixed-nat-freed       Fixed-NAT freed
  port-allocated        Port allocated
  port-batch-allocated  Port Batch allocated
  port-batch-freed      Port Batch freed
  port-freed            Port freed
```

Continue to drill down into the keywords to determine which events are available:

```
AX#show ip nat logging keywords lsn port-freed
  $proto-name$          Protocol name
  $proto-num$           Protocol number
  $src-ip$              Source IP
  $src-port$            Source port
  $nat-ip$              NAT IP
  $nat-port$            NAT Port
```

Therefore, to configure a custom message string for CGN when a port is freed, use the following command at the configuration level for the logging template:

```
AX(config)#ip nat template logging LSN_LOG
AX(config-nat logging)#rfc-custom message lsn port-freed "CGN:Port-Freed [$src-ip$
$src-port$ $nat-ip$ $nat-port$]"
```

***Note:** The message string must be encapsulated within “ ” and may have text embedded within the string. The events must be encapsulated within [ ]. Refer to the “AX Series IPv4-to-IPv6 Transition Solutions Guide” and RFC 5424 for more details.*

5.11.11 CGN LOGGING TO SYSLOG OVER TCP

The AX devices support syslog over TCP to provide reliable log message transport. Configuring TCP logging is identical to configuring UDP logging, with the exception of the server and service group configuration. To configure TCP logging, follow these steps.

1. At the global configuration level, add a log server to the configuration. A name and the IP address of the server must be specified.

   ```
   AX(config)#slb server syslog1 10.10.10.11
   ```

2. At the real server configuration level, specify the TCP port and protocol for the syslog service.

   ```
   AX(config-real server)#port 601 tcp
   AX(config-real server-node port)#exit
   AX(config-real server)#exit
   ```

3. At the global configuration level, create the service group. Specify the group name and TCP protocol.

   ```
   AX(config)#slb service-group syslog tcp
   ```

4. Add the member to the group. Specify the server name given in step 1 and the port number from step 2.

   ```
   AX(config-slb svc group)#member syslog1:601
   AX(config -slb svc group)#exit
   ```

5. Create the logging template and specify the syslog server group and the events to log.

   ```
   AX(config)#ip nat template logging LSN_LOG
   AX(config-nat logging)#log sessions
   AX(config-nat logging)#log fixed-nat-all
   AX(config-nat logging)#log fixed-nat-user-ports
   AX(config-nat logging)#service-group syslog
   AX(config-nat logging)#exit
   ```

***Note:** In this example, the service group name is "syslog", and both CGN events (log sessions) and Fixed-NAT events are logged. Alternatively, logging formats, source port for syslog, and so on, also can be modified at this configuration level. Refer to the “AX Series IPv4-to-IPv6 Transition Solutions Guide” for more information.*

6. Activate the template by committing the following command at the global configuration level, and specifying the template name given in step 3.

   ```
   AX(config)#ip nat lsn logging default-template LSN_LOG
   ```

7. Verify operation using the following command:

   ```
   AX(config)#show ip nat logging statistics debug
   NAT Logging Statistics:
   ----------------------------
   TCP Session Created                      7183
   TCP Session Deleted                      7178
   TCP Port Allocated                       4499
   TCP Port Freed                           4493
   TCP Port Batch Allocated                 40
   TCP Port Batch Freed                     40
   UDP Session Created                      88018
   UDP Session Deleted                      87392
   UDP Port Allocated                       156
   UDP Port Freed                           187
   UDP Port Batch Allocated                 39
   UDP Port Batch Freed                     39
   ICMP Session Created                     1
   ICMP Session Deleted                     1
   ICMP Resource Allocated                  1
   ICMP Resource Freed                      1
   ICMPV6 Session Created                   0
   ICMPV6 Session Deleted                   0
   ICMPV6 Resource Allocated                0
   ICMPV6 Resource Freed                    0
   GRE Session Created                      0
   GRE Session Deleted                      0
   GRE Resource Allocated                   0
   GRE Resource Freed                       0
   ESP Session Created                      0
   ESP Session Deleted                      0
   ESP Resource Allocated                   0
   ```

```
        ESP Resource Freed                       0
        Fixed NAT Inside User Port Mapping       0
        Fixed NAT Disabled Configs Logged        0
        Fixed NAT Disabled Config Logs Sent      0
        Log Packets Sent                         0
        Log Packets Dropped                      0
        Tcp Connection Established               5
        Tcp Connection Lost                      0
        TCP Port Overloading Allocated           0
        TCP Port Overloading Freed               0
        UDP Port Overloading Allocated           0
        UDP Port Overloading Freed               0
        Out of Buffers                           0
        Add Message to Buffer Failed             0
        RTSP UDP Port Allocated                  0
        RTSP UDP Port Freed                      0
        Tcp Connection Failed                    0
```

The AX device establishes TCP connections from interface IP addresses to the syslog servers and maintains them. These sessions are visible in the session table and have a forwarding source/destination of 0.0.0.0.

```
AX(config)#show session | include 10.10.10.11
Prot Forward Source   Forward Dest    Reverse Source    Reverse Dest    Age   Hash Flags
---------------------------------------------------------------------------------------
Tcp  0.0.0.0        0.0.0.0     10.10.10.11:601    10.10.10.10:20040      120   1    NS
Tcp  0.0.0.0        0.0.0.0     10.10.10.11:601    10.10.10.10:20041      240   2    NS
Tcp  0.0.0.0        0.0.0.0     10.10.10.11:601    10.10.10.10:20042      240   3    NS
Tcp  0.0.0.0        0.0.0.0     10.10.10.11:601    10.10.10.10:20043      420   4    NS
Tcp  0.0.0.0        0.0.0.0     10.10.10.11:601    10.10.10.10:20044      120   5    NS
Total Sessions:          5
```

5.11.12 CGN LOGGING TO RADIUS

The AX Series supports CGN logging to RADIUS. The AX device acts as a RADIUS client and provides identical logging events as the Syslog implementation using RADIUS Accounting-Request messages. CGN logging to RADIUS provides a trusted logging environment and simplifies log message analysis.

To configure CGN logging to RADIUS, follow these steps:

1. At the global configuration level, add a log server to the configuration. A name and the IP address of the server must be specified.

```
AX(config)#slb server radius1 10.10.10.11
```

2. At the config-real server configuration level, specify the RADIUS UDP port and protocol for the syslog service.

```
AX(config-real server)#port 1813 udp
AX(config-real server-node port)#exit
AX(config-real server)#exit
```

3. At the global configuration level, create the service group. Specify the group name and TCP protocol.

```
AX(config)#slb service-group radiusgp udp
```

4. Add the member to the group. Specify the server name and UDP protocol.

```
AX(config-slb svc group)#member radius1:1813
AX(config -slb svc group)#exit
```

5. Create the logging template and specify the RADIUS server group, RADIUS secret, and the events to log.

```
AX(config)#ip nat template logging LSN_LOG
AX(config-nat logging)#log sessions
AX(config-nat logging)#log-receiver radius secret a10rad
AX(config-nat logging)#service-group syslog
AX(config-nat logging)#exit
```

Note: *In this example, the service group name is "radiusgp", the secret is "a10rad", and CGN events (log sessions) are logged. Note that for RADIUS logging, the* ***source-port****,* ***format****,* ***rfc-custom****,* ***facility*** *and* ***severity*** *options do not apply within the configuration context. Please consult the “AX Series IPv4-to-IPv6 Transition Solutions Guide” for more information.*

6. Activate the template by committing the following command at the global configuration level, specifying the template name given in step 3.

```
AX(config)#ip nat lsn logging default-template LSN_LOG
```

7. Verify there are no packets dropped to server. If the counter is incrementing, verify at least one RADIUS server is up:

```
AX(config)#show ip nat logging statistics debug


NAT Logging Statistics:
----------------------------
TCP Session Created                      7183
TCP Session Deleted                      7178
TCP Port Allocated                       4499
TCP Port Freed                           4493
TCP Port Batch Allocated                 40
TCP Port Batch Freed                     40
UDP Session Created                      88018
UDP Session Deleted                      87392
UDP Port Allocated                       156
UDP Port Freed                           187
UDP Port Batch Allocated                 39
UDP Port Batch Freed                     39
ICMP Session Created                     1
ICMP Session Deleted                     1
ICMP Resource Allocated                  1
ICMP Resource Freed                      1
ICMPV6 Session Created                   0
ICMPV6 Session Deleted                   0
ICMPV6 Resource Allocated                0
ICMPV6 Resource Freed                    0
GRE Session Created                      0
GRE Session Deleted                      0
GRE Resource Allocated                   0
GRE Resource Freed                       0
ESP Session Created                      0
ESP Session Deleted                      0
ESP Resource Allocated                   0
ESP Resource Freed                       0
Fixed NAT Inside User Port Mapping       0
Fixed NAT Disabled Configs Logged        0
```

```
        Fixed NAT Disabled Config Logs Sent      0
        Log Packets Sent                         0
        Log Packets Dropped                      6
        Tcp Connection Established               0
        Tcp Connection Lost                      0
        TCP Port Overloading Allocated           0
        TCP Port Overloading Freed               0
        UDP Port Overloading Allocated           0
        UDP Port Overloading Freed               0
        Out of Buffers                           0
        Add Message to Buffer Failed             0
        RTSP UDP Port Allocated                  0
        RTSP UDP Port Freed                      0
        Tcp Connection Failed                    0
```

Like Syslog over TCP, RADIUS logging will place an entry in the session table for the logging server with forwarding source/destination of 0.0.0.0:

```
AX(config)#show session | include 0.0.0.0
Prot Forward Source   Forward Dest    Reverse Source    Reverse Dest   Age    Hash Flags
--------------------------------------------------------------------------------------
UDP  0.0.0.0            0.0.0.0      10.10.10.11:1813    10.10.10.10:20040   120   1    NS
```

5.11.13 LOG BATCHING

By default, the AX CGN device sends multiple log messages per packet to the external logging server. In some cases, a particular syslog implementation may not handle this situation correctly.

For deployments where logging traffic is minimal or proper syslog operation is not occurring, disable log batching at the configuration level for the logging template.

For example:

```
AX(config)#ip nat template logging LSN_LOG
AX(config-nat logging)#batched-logging-disable
```

5.11.14 PRECISION TIME STAMP

The AX Series provides a logging option that increases the precision of the log timestamps. By default, log message timestamps are precise to within 1 whole second. With precision timestamps enabled, log message timestamps are precise to within 1/100th of a second. Precision timestamps are supported for CGN logging to both Syslog and RADIUS using Binary and Hex formatting.

To enable precision timestamps, use the **resolution** command at the configuration level for the logging template.

For example:

```
AX(config)#ip nat template logging LSN_LOG
AX(config-nat logging)#resolution 10-milliseconds
```

5.11.15 NAT POOL LOGGING TEMPLATE ASSIGNMENT

The default behavior for CGN logging is to use the default template for all CGN pools. Recall that the template controls the following logging attributes:

- NAT logging facility
- Format
- Events to be logged
- Timestamp resolution
- Target server for logging
- Log method (RADIUS, syslog, and so on)
- Source port

Some environments may require the flexibility to enable different logging parameters per NAT pool. The AX device supports the ability to map an individual NAT pool to a logging template. This allows specific logging attributes to be assigned with more granularity. Every pool that is not specifically assigned to a logging template will use the default template.

To enable this feature, use the following command at the global configuration level:

```
ip nat lsn logging pool pool-name template logging-template-name
```

For example:

```
AX(config)#ip nat lsn logging pool test_pool template LSN_LOG_2
```

6 NETWORK INTEGRATION

This section provides information and best practices for integrating the AX Series CGN device into static and dynamically routed environments.

6.1 STATIC ROUTE DEPLOYMENT

The AX Series supports all major routing protocols, providing a flexible framework that integrates into networking environments with minimal disruption. Alternatively, some service providers may choose to use static routing from access networks to the AX device and outwards towards the Internet. Referring to Figure 2, notice that the access network layer and external layers are naturally separated by the AX device. In this case, any access network aggregation routers must have a static route to the AX device and the AX device must have a default route to the external peering router and a static route to the access network. The external routers must have a static route to the NAT Pool IP subnets. Using HA, the floating IP addresses between the HA pair should be the next hop for the statically defined routes.

For the example in Figure 2, the following static routes should be configured:

- Access routers should have a default route to the AX floating IP address, 100.64.1.1.
- The AX device should have a static route to 100.64.100/24 and 100.64.101/24, with next hop 100.64.1.4 (the IP address of the access router).
- The AX device should have a default route to the external peering router, 10.200.2.10.
- The external router should have a route to NAT pool 192.0.2.32/27, with next hop 10.200.2.1 (the floating IP address for the 10.200.2.0 subnet).

The AX device also can apply gateway health checks to optimize selection of the active HA device and ensure that the active AX device always can reach both downstream and upstream routers. If the gateway health check for a particular router fails, the HA process declares the device down and the standby AX assumes active status.

To configure gateway health checks, use the following configuration steps:

1. Create a health monitor and assign ICMP health check with name “ext_peer”.

```
AX(config)#health monitor ext_peer
AX(config-health:monitor)#method icmp
AX(config-health:monitor)#exit
```

2. Configure the real server and apply the health monitor to it.

```
AX(config)#health monitor ext_peer
AX(config-health:monitor)#exit
AX2(config)#slb server gateway1 10.200.2.10
AX(config-real server)#health-check ext_peer
AX(config-real server)#exit
```

3. Enable HA health monitoring for the gateway.

```
AX(config)#ha check gateway 10.200.2.10
```

6.2 DYNAMIC ROUTING

The example in Figure 2 uses dynamic routing for reachability. BGP is enabled between the external peering router and the AX device, and OSPF is enabled between the AX device and the access router. While this provides basic connectivity, it does not provide the floating IP or NAT pool redistribution that is required for proper operation of the CGN application.

The following additional options should be configured to enable route redistribution.

1. Configure the upstream router to originate default route(s) towards the active and standby AX devices.

```
AX(config)#router bgp 65000
AX(config-router:device1)#neighbor 10.200.2.2 remote-as 65000
AX(config-router:device1)#neighbor 10.200.2.2 default-originate
AX(config-router:device1)#neighbor 10.200.2.3 remote-as 65000
AX(config-router:device1)#neighbor 10.200.2.3 default-originate
AX(config-router:device1)#route-map nat_redis permit 1
AX(config-route-map)#set ip next-hop 10.200.2.1
AX(config-route-map)#exit
```

2. Configure the AX device to redistribute the NAT pool through BGP to the peering router, 10.200.2.10. Modify the next hop to the floating IP address. Create a route map to set the next hop to the floating IP, 10.200.2.1. Use the **redistribute nat** command (at the BGP configuration level) to distribute the NAT pool to all BGP peers.

```
AX(config)#router bgp 65000
AX(config-router:device1)#redistribute ip-nat route-map nat_redis
AX(config-router:device1)#exit
```

3. Configure the AX device to send the default route through OSPF to all downstream routers. The next hop must be modified to the floating IP address. Create a route map to set the next hop to the floating IP address, 100.64.1.1.

```
AX(config)#router ospf 1
AX(config-router:device1)#default-information originate always route-map
default_route
AX(config-router:device1)#route-map default_route permit 1
AX(config-route-map)#set ip next-hop 100.64.1.1
```

6.3 HA CONSIDERATIONS

This section describes some additional considerations if you use HA.

6.3.1 FLOATING IP ADDRESS REDISTRIBUTION

Although not required in the example illustrated in Figure 2, some networking environments may require redistribution of the floating IP addresses to support AX device reachability.

For example, to configure OSPF to redistribute a floating IP address, use the following command at the configuration level for the OSPF instance:

```
redistribute floating-ip
```

This provides basic redistribution of the floating IP address. In addition, metrics and OSPF tags can be modified, and route maps can be applied for more granularity.

6.3.2 HA STANDBY ROUTER

When HA is implemented on the AX Series, the active router advertises and receives IP prefixes for all routing protocols. The standby device implements a different behavior. For BGP, the standby device will establish and maintain the BGP connection, but it will not inject routes to any peers. This behavior eliminates the issue of packets being redirected towards the standby device and any subsequent packet loss.

Since link state protocols require that every device contain an identical database, a different approach is needed for OSPF. An additional cost can be assigned to an AX device's OSPF interfaces when the HA status for any group on the device is Standby. If failover of one or more HA groups from Active to Standby occurs, the AX device updates the cost of all its OSPF interfaces and sends Link-State Advertisement (LSA) updates to its OSPF neighbors advertising the interface cost change. After an OSPF neighbor receives the LSA update, the neighbor updates its OSPF link-state database with the increased cost of the links. The increased cost biases route selection away from paths that use the standby AX device.

To enable OSPF awareness of HA, use the following command at the OSPF configuration level:

```
ha-standby-extra-cost num
```

The cost can be 1-65535.

6.3.3 ACTIVE-ACTIVE OPERATION

The AX CGN device allows for active-active configuration for HA. The HA implementation provides up to 31 groups, each supporting independent active/standby parameters. An AX CGN device can act as both active and standby for multiple groups, allowing load sharing between the HA pair; however, it can only act as either active or standby per group.

For example, the operator could configure two HA groups, 1 and 2, and specify AX-1 to be active for group 1 (standby for group 2) and AX-2 to be active for group 2 (standby for group 1). In this case, there will be a floating IP address for each group that acts as the default route for downstream devices. The use of either BGP or OSPF (along with the **ha-standby-extra-cost** option) and policy-based-routing in the aggregation layers, will ensure that traffic is always sent to the active device for the HA group.

7 BEST PRACTICES

This section provides a summary of the recommendations and best practices presented throughout this guide.

- Deploy CGN devices in “one-arm-router” topologies as shown in Figure 2. This adds migration flexibility, increases scale, and facilitates moves, additions, and changes.
- Link aggregation can be used for link/optical redundancy between aggregation and CGN devices.
- Assign IP addresses to VE interfaces instead of physical interfaces, to allow for future configuration flexibility.
- CGN devices should be deployed in HA pairs for seamless reliability.
- HA connection mirroring should be enabled to ensure connection continuity.
- HA links should be dedicated, if possible. This isolates sensitive information, mitigating potential security threats, and reducing link utilization and congestion.
- Floating IP addresses (virtual IP address shared between HA pairs) should be configured for downstream and upstream traffic. Any access or peering equipment directly attached to the CGN devices should set their default gateways to the appropriate floating IP address.

- If the environment is statically routed, then gateway health checks should be utilized to qualify default gateways and minimize packet loss. Upstream devices must include a static route for reachability to the configured NAT pool address space.

- If CGN is deployed in a dynamically routed environment, peering routers should originate a default route and redistribute towards the CGN devices. Likewise, the CGN devices must redistribute the NAT pool address space to the upstream routers and modify the next hop to the floating IP address. CGN devices also must originate a default route downstream for the access devices and modify the next hop to their floating IP address.

- OSPF metrics should be increased on the standby CGN device. The AX Series supports OSPF increased cost dynamically within the HA architecture, thus giving OSPF awareness of the HA state, and ensuring traffic is not sent to the standby device.

- The use of pool groups adds flexibility and eases configuration steps for NAT pool changes. Example:

```
ip nat pool POOL1 88.88.88.1 88.88.88.126 netmask /25  ha-group-id 1 lsn
ip nat pool POOL2 99.99.99.129 99.99.99.254 netmask /25  ha-group-id 1 lsn
ip nat pool-group POOL-GROUP1 POOL1 POOL2
```

- Enable all required ALGs. FTP is enabled by default.

- In deployments where logging traffic is minimal, or issues are seen with proper syslog operation, disable the log batching feature.

- The AX Series does not default to the device’s maximum session allocations or NAT resource allocations. Adjust these system parameters as appropriate for the individual deployment.

- If minimal logging is required, or log file resources are minimal, deterministic or Fixed-NAT should be deployed. Fixed-NAT allocates a dedicated block of ports to each inside IP address, negating the need for logging. It also has the flexibility to provide minimal logging in environments where external logging is required for legal or corporate policy compliance.

- The AX Series supports EIM/EIF (full-cone behavior), address pooling, port preservation, and hairpinning by default. It is recommended that these features remain enabled to ensure uninterrupted service for peer-to-peer applications.

- The STUN timeout determines the timeout period for full-cone sessions. The default is 2 minutes. Some applications may require an increased timeout value. The supported values are 0-60 minutes.

- For environments that do not require peer-to-peer application support and have minimal NAT IP addresses, protocol port overloading can be used to increase efficiency of the NAT IP address allocations.

- When using protocol port overloading, an IP/port translation for one client can be re-used for multiple clients, providing even more usage of a single IP address. The drawback is that tracking the flow from each client to the server is not possible anymore.
- DNS supports fast session aging, resulting in a 3-second timeout. Some environments may dictate a longer timeout period and require appropriate configuration.
- To tightly control system resources, session limits and user quotas should be configured.

8 SUMMARY AND CONCLUSION

The configuration example in this Deployment Guide shows how to set up a basic CGN deployment including connectivity to the Internet. A10’s CGN solution has numerous configurable options, some of which are described in the advanced configuration section. The CGN feature set on the AX Series provides the following key advantages:

- Transparent NAT connectivity through EIM/EIF
- Inter-connectivity through hairpinning and interplay
- Fairness and resource sharing
- Comprehensive logging options

The AX Series provides a feature-rich, powerful and cost-effective platform for implementing Carrier Grade NAT.