BIG-IP® CGNAT: Implementations

Version 11.4

Table of Contents

Legal Notices

Publication Date

This document was published on May 15, 2013.

Publication Number

MAN-0428-01

Copyright

Trademarks

Access Policy Manager, Advanced Client Authentication, Advanced Routing, APM, Application Security Manager, ARX, AskF5, ASM, BIG-IP, BIG-IQ, Cloud Extender, CloudFucious, Cloud Manager, Clustered Multiprocessing, CMP, COHESION, Data Manager, DevCentral, DevCentral [DESIGN], DNS Express, DSC, DSI, Edge Client, Edge Gateway, Edge Portal, ELEVATE, EM, Enterprise Manager, ENGAGE, F5, F5 [DESIGN], F5 Management Pack, F5 Networks, F5 World, Fast Application Proxy, Fast Cache, FirePass, Global Traffic Manager, GTM, GUARDIAN, IBR, Intelligent Browser Referencing, Intelligent Compression, IPv6 Gateway, iApps, iControl, iHealth, iQuery, iRules, iRules OnDemand, iSession, L7 Rate Shaping, LC, Link Controller, Local Traffic Manager, LTM, Message Security Manager, MSM, OneConnect, OpenBloX, OpenBloX [DESIGN], Packet Velocity, Policy Enforcement Manager, PEM, Protocol Security Manager, PSM, Real Traffic Policy Builder, Rosetta Diameter Gateway, Scale^{N}, Signaling Delivery Controller, SDC, SSL Acceleration, StrongBox, SuperVIP, SYN Check, TCP Express, TDR, TMOS, Traffic Management Operating System, Traffix Diameter Load Balancer, Traffix Systems, Traffix Systems (DESIGN), Transparent Data Reduction, UNITY, VAULT, VIPRION, vCMP, virtual Clustered Multiprocessing, WA, WAN Optimization Manager, WebAccelerator, WOM, and ZoneRunner, are trademarks or service marks of F5 Networks, Inc., in the U.S. and other countries, and may not be used without F5's express written consent.

All other product and company names herein may be trademarks of their respective owners.

Export Regulation Notice

This product may include cryptographic software. Under the Export Administration Act, the United States government may consider it a criminal offense to export this product from the United States.

RF Interference Warning

This is a Class A product. In a domestic environment this product may cause radio interference, in which case the user may be required to take adequate measures.

FCC Compliance

This equipment has been tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of FCC rules. These limits are designed to provide reasonable protection against harmful

interference when the equipment is operated in a commercial environment. This unit generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

Any modifications to this device, unless expressly approved by the manufacturer, can void the user's authority to operate this equipment under part 15 of the FCC rules.

Canadian Regulatory Compliance

This Class A digital apparatus complies with Canadian ICES-003.

Standards Compliance

This product conforms to the IEC, European Union, ANSI/UL and Canadian CSA standards applicable to Information Technology products at the time of manufacture.

Acknowledgments

This product includes software developed by Bill Paul.

This product includes software developed by Jonathan Stone.

This product includes software developed by Manuel Bouyer.

This product includes software developed by Paul Richards.

This product includes software developed by the NetBSD Foundation, Inc. and its contributors.

This product includes software developed by the Politecnico di Torino, and its contributors.

This product includes software developed by the Swedish Institute of Computer Science and its contributors.

This product includes software developed by the University of California, Berkeley and its contributors.

This product includes software developed by the Computer Systems Engineering Group at the Lawrence Berkeley Laboratory.

This product includes software developed by Christopher G. Demetriou for the NetBSD Project.

This product includes software developed by Adam Glass.

This product includes software developed by Christian E. Hopps.

This product includes software developed by Dean Huxley.

This product includes software developed by John Kohl.

This product includes software developed by Paul Kranenburg.

This product includes software developed by Terrence R. Lambert.

This product includes software developed by Philip A. Nelson.

This product includes software developed by Herb Peyerl.

This product includes software developed by Jochen Pohl for the NetBSD Project.

This product includes software developed by Chris Provenzano.

This product includes software developed by Theo de Raadt.

This product includes software developed by David Muir Sharnoff.

This product includes software developed by SigmaSoft, Th. Lockert.

This product includes software developed for the NetBSD Project by Jason R. Thorpe.

This product includes software developed by Jason R. Thorpe for And Communications, http://www.and.com.

This product includes software developed for the NetBSD Project by Frank Van der Linden.

This product includes software developed for the NetBSD Project by John M. Vinopal.

This product includes software developed by Christos Zoulas.

This product includes software developed by the University of Vermont and State Agricultural College and Garrett A. Wollman.

This product includes software developed by Balazs Scheidler (bazsi@balabit.hu), which is protected under the GNU Public License.

This product includes software developed by Niels Mueller (nisse@lysator.liu.se), which is protected under the GNU Public License.

In the following statement, *This software* refers to the Mitsumi CD-ROM driver: This software was developed by Holger Veit and Brian Moore for use with 386BSD and similar operating systems. *Similar operating systems* includes mainly non-profit oriented systems for research and education, including but not restricted to NetBSD, FreeBSD, Mach (by CMU).

This product includes software developed by the Apache Group for use in the Apache HTTP server project (http://www.apache.org/).

This product includes software licensed from Richard H. Porter under the GNU Library General Public License (© 1998, Red Hat Software), www.gnu.org/copyleft/lgpl.html.

This product includes the standard version of Perl software licensed under the Perl Artistic License (© 1997, 1998 Tom Christiansen and Nathan Torkington). All rights reserved. You may find the most current standard version of Perl at http://www.perl.com.

This product includes software developed by Jared Minch.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

This product contains software based on oprofile, which is protected under the GNU Public License.

This product includes RRDtool software developed by Tobi Oetiker (http://www.rrdtool.com/index.html) and licensed under the GNU General Public License.

This product contains software licensed from Dr. Brian Gladman under the GNU General Public License (GPL).

This product includes software developed by the Apache Software Foundation (http://www.apache.org/).

This product includes Hypersonic SQL.

This product contains software developed by the Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, and others.

This product includes software developed by the Internet Software Consortium.

This product includes software developed by Nominum, Inc. (http://www.nominum.com).

This product contains software developed by Broadcom Corporation, which is protected under the GNU Public License.

This product contains software developed by MaxMind LLC, and is protected under the GNU Lesser General Public License, as published by the Free Software Foundation.

This product includes Intel QuickAssist kernel module, library, and headers software licensed under the GNU General Public License (GPL).

This product includes software licensed from Gerald Combs (gerald@wireshark.org) under the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or any later version. Copyright ©1998 Gerald Combs.

This product includes software developed by Thomas Williams and Colin Kelley. Copyright ©1986 - 1993, 1998, 2004, 2007

Permission to use, copy, and distribute this software and its documentation for any purpose with or without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Permission to modify the software is granted, but not the right to distribute the complete modified source code. Modifications are to be distributed as patches to the released version. Permission to distribute binaries produced by compiling modified sources is granted, provided you

1. distribute the corresponding source modifications from the released version in the form of a patch file along with the binaries,

2. add special version identification to distinguish your version in addition to the base release version number,
3. provide your name and address as the primary contact for the support of your modified version, and
4. retain our contact information in regard to use of the base software.

Permission to distribute the released version of the source code along with corresponding source modifications in the form of a patch file is granted with same provisions 2 through 4 for binary distributions. This software is provided "as is" without express or implied warranty to the extent permitted by applicable law.

This product contains software developed by Google, Inc. Copyright ©2011 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Chapter 1

Deploying a Carrier Grade NAT

About the carrier-grade NAT (CGNAT) module

The carrier-grade network address translation (CGNAT) module on the BIG-IP® system supports large groups of translation addresses using large-scale NAT (LSN) pools and grouping of address-translation-related options in an ALG profile, which can be assigned to multiple virtual servers. It also has the ability to match virtual servers based on client address to destination addresses and ports. Other characteristics of the CGNAT module are listed here.

Translation address persistence

The CGNAT module can assign the same external (translation) address to all connections originated by the same internal client. For example, providing endpoint-independent address mapping.

Automatic external inbound connection handling

CGNAT can accept inbound external connections to active translation address/port combinations to facilitate endpoint-independent filtering as described in section 5 of *RFC 4787*. This is also known as a full-cone NAT.

More efficient logging

Log messages that map external addresses and ports back to internal clients for troubleshooting and law enforcement/legal compliance are supported.

Deterministic assignment of translation addresses

Deterministic mode is an option to assign translation address and port based on the client address/port and destination address/port. It uses reversible mapping to reduce logging, while maintaining translated IP address discoverability for troubleshooting and law compliance. Deterministic mode also provides an option to configure backup-members.

Licensing

Geared toward service providers, the CGNAT module is offered as a stand-alone license or as an add-on license for Local Traffic Manager™ (LTM®) and Policy Enforcement Manager (PEM).

About ALG Profiles

Application Layer Gateway (ALG) profiles provide the CGNAT with enough protocol and service knowledge to carry out the necessary application protocol header and payload modifications that allow these protocols to seamlessly traverse the NAT. FTP, RSTP, and SIP profiles are supported with ALG profiles and may be added to the CGNAT configuration as needed.

Important: *ALG traffic cannot use a deterministically-mapped address. Using a separate NAPT pool for these translations is recommended.*

Task summary

Perform these tasks to deploy a source translation using CGNAT.

Creating an LSN pool

The CGNAT module must be enabled through **System** > **Resource Provisioning** before LSN pools can be configured.

Large Scale NAT (LSN) pools are used by the CGNAT module to allow efficient configuration of translation prefixes and parameters.

1. On the Main tab, click **Carrier Grade NAT** > **LSN Pools**.
   The LSN Pool List screen opens.
2. Click **Create**.
3. Enter a unique name in the **Name** field.
4. In the Configuration area, for the **Member List** setting, enter an address and a prefix length in the **Address/Prefix Length** field and click **Add**.

   If your pool uses deterministic mode, ensure that any address ranges you enter as a member do not overlap another member's prefix address ranges. For example, the address and prefix `10.10.10.0/24` overlaps `10.10.10.0/23`.
5. Click **Finished**.

Your LSN pool is now ready and you can continue to configure your CGNAT.

Configuring a SIP ALG profile

You must have a SIP registrar and proxy configured prior to using a SIP ALG profile.

The SIP ALG profile provides the CGNAT module with enough protocol and service knowledge to make specified packet modifications to the IP and TCP/UDP headers, as well as the SIP payload during translation.

Important: *Only edit copies of the included ALG profiles to avoid unwanted propagation of settings to other profiles that use the included profiles as parents.*

---

1. On the Main tab, click **Carrier Grade NAT** > **ALG Profiles** > **SIP**.
   The SIP screen opens and displays a list of available SIP ALG profiles.
2. Click **Create** to open the New SIP Profile screen.
3. Enter a name for the new profile.
4. From the **Parent Profile** list, select **sip** as the new profile.
5. For the **Terminate on BYE** setting, select the **Enabled** check box.
6. Select the **Dialog Aware** check box, and enter a unique community string in the **Community** field.
7. From the **Insert Via Header** list, select **Enabled**.
8. Click **Finished** to save the new SIP ALG profile.
9. You must also create two virtual servers: one to handle SIP TCP traffic and another to handle SIP UDP traffic.

a) Create a host virtual server with a **Source** address of `0.0.0.0/0` and a **Destination** type set as **Network**, as well as a **Mask** of `0.0.0.0` and a **Service Port** of `5060`.
b) From the **Protocol** list, select **TCP**.
c) From the **SIP Profile** list, select a SIP profile.
d) From the **VLAN and Tunnel Traffic** list, select **All VLANs and Tunnels**.
e) From the **LSN Pool** list, select an LSN pool.
f) Repeat the virtual server creation procedure, and then from the **Protocol** list, choose **UDP**. Also choose the SSL client, SSL server, and Authentication profiles from their respective lists as needed.

You now have a TCP and UDP virtual server to handle SIP traffic.

You now have a SIP ALG profile for use by CGNAT.

Configuring a CGNAT iRule

You create iRules® to automate traffic forwarding for XML content-based routing. When a match occurs, an iRule event is triggered, and the iRule directs the individual request to an LSN pool, a node, or virtual server.

1. On the Main tab, click **Carrier Grade NAT** > **iRules**.
   The iRule List screen opens.
2. Click **Create**.
3. In the **Name** field, type a 1 to 31 character name, such as `cgn_https_redirect_iRule`.
4. In the **Definition** field, type the syntax for the iRule using Tool Command Language (Tcl) syntax.
   For complete and detailed information iRules syntax, see the F5 Networks DevCentral web site (`http://devcentral.f5.com`).
5. Click **Finished**.

You now have an iRule to use with a CGNAT virtual server.

Creating a virtual server for an LSN pool

Virtual servers are matched based on source (client) addresses. Define a virtual server that references the CGNAT profile and the LSN pool.

1. On the Main tab, click **Carrier Grade NAT** > **Virtual Servers**.
   The Virtual Servers screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. From the **Type** list, select **Performance (Layer 4)**.
5. For the **Destination** setting, in the **Address** field, type `0.0.0.0` to allow all traffic to be translated.
6. In the **Service Port** field, type `*` or select *** All Ports** from the list.
7. From the **VLAN and Tunnel Traffic** list, select **Enabled on**. Then, for the **VLANs and Tunnels** setting, move the VLAN or VLANs on which you want to allow the virtual servers to share traffic from the **Available** list to the **Selected** list.
8. For the **LSN Pool** setting, select the pool that this server will draw on for translation addresses.

9. In the Resources area of the screen, for the **iRules** setting, select the name of the iRule that you want to assign and using the Move button, move the name from the **Available** list to the **Enabled** list.
10. Click **Finished**.

The custom CGNAT virtual server now appears in the CGNAT Virtual Servers list.

Creating a CGNAT tunnel

Many translations use tunneling to move TCP/UDP traffic where the payload is other IP traffic. Create and configure a tunnel for use with an LSN pool.

1. On the Main tab, click **Carrier Grade NAT** > **Tunnels**.
   The Tunnels screen opens.
2. Click **Create**.
   The New Tunnel screen opens.
3. In the **Name** field, type a unique name for the tunnel.
4. In the **Local Address** field, type the IP address of the BIG-IP system.
5. From the **Remote Address** list, select **Specify**, and type a wildcard address (`::` or `0.0.0.0`) as the other end of the tunnel.
6. Click **Finished**.

Your CGNAT tunnel is ready to use as an egress interface in an LSN Pool.

Chapter 2

Using NAT64 to Map IPv6 Addresses to IPv4 Destinations

About NAT64

For the BIG-IP® system CGNAT module, NAT64 is the NAT type that maps IPv6 subscriber private addresses to IPv4 Internet public addresses. NAT64 translates subscriber IPv6 addresses to public Internet IPv4 addresses and allows Internet traffic from an IPv6 client to reach a public IPv4 server. The CGNAT module processes NAT64 traffic, as defined in *RFC 6146* for TCP and UDP addresses.

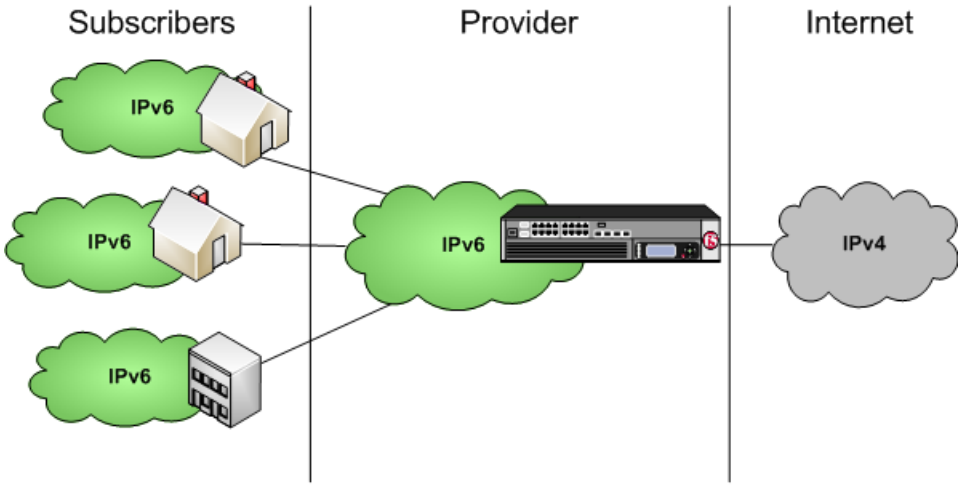

Figure 1: Diagram of a NAT64 network

Task summary

Perform these tasks to use NAT64 to map IPv6 addresses to IPv4 destinations.

Creating a NAT64 LSN pool
Creating a virtual server for an LSN pool
Configuring a SIP ALG profile
Configuring a CGNAT iRule

Creating a NAT64 LSN pool

The CGNAT module must be enabled through **System** > **Resource Provisioning** before LSN pools can be configured.

A NAT64 LSN pool contains the set of IPv4 address ranges that will be used on the public Internet.

1. On the Main tab, click **Carrier Grade NAT** > **LSN Pools**.
   The LSN Pool List screen opens.
2. Click **Create**.
3. Enter a unique name in the **Name** field.
4. Enter an address and a prefix length in **Address/Prefix Length** and click **Add**.
5. Click **Finished**.

Your LSN pool is now ready and you can continue to configure your CGNAT.

Creating a virtual server for an LSN pool

Virtual servers are matched based on source (client) addresses. Define a virtual server that references the CGNAT profile and the LSN pool.

1. On the Main tab, click **Carrier Grade NAT** > **Virtual Servers**.
   The Virtual Servers screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. From the **Type** list, select **Performance (Layer 4)**.
5. For the **Destination** setting, in the **Address** field, type `0.0.0.0` to allow all traffic to be translated.
6. In the **Service Port** field, type `*` or select *** All Ports** from the list.
7. From the **VLAN and Tunnel Traffic** list, select **Enabled on**. Then, for the **VLANs and Tunnels** setting, move the VLAN or VLANs on which you want to allow the virtual servers to share traffic from the **Available** list to the **Selected** list.
8. For the **LSN Pool** setting, select the pool that this server will draw on for translation addresses.
9. In the Resources area of the screen, for the **iRules** setting, select the name of the iRule that you want to assign and using the Move button, move the name from the **Available** list to the **Enabled** list.
10. Click **Finished**.

The custom CGNAT virtual server now appears in the CGNAT Virtual Servers list.

Configuring a SIP ALG profile

You must have a SIP registrar and proxy configured prior to using a SIP ALG profile.

The SIP ALG profile provides the CGNAT module with enough protocol and service knowledge to make specified packet modifications to the IP and TCP/UDP headers, as well as the SIP payload during translation.

Important: *Only edit copies of the included ALG profiles to avoid unwanted propagation of settings to other profiles that use the included profiles as parents.*

---

1. On the Main tab, click **Carrier Grade NAT** > **ALG Profiles** > **SIP**.
   The SIP screen opens and displays a list of available SIP ALG profiles.
2. Click **Create** to open the New SIP Profile screen.
3. Enter a name for the new profile.
4. From the **Parent Profile** list, select **sip** as the new profile.
5. For the **Terminate on BYE** setting, select the **Enabled** check box.

6. Select the **Dialog Aware** check box, and enter a unique community string in the **Community** field.
7. From the **Insert Via Header** list, select **Enabled**.
8. Click **Finished** to save the new SIP ALG profile.
9. You must also create two virtual servers: one to handle SIP TCP traffic and another to handle SIP UDP traffic.
   a) Create a host virtual server with a **Source** address of `0.0.0.0/0` and a **Destination** type set as **Network**, as well as a **Mask** of `0.0.0.0` and a **Service Port** of `5060`.
   b) From the **Protocol** list, select **TCP**.
   c) From the **SIP Profile** list, select a SIP profile.
   d) From the **VLAN and Tunnel Traffic** list, select **All VLANs and Tunnels**.
   e) From the **LSN Pool** list, select an LSN pool.
   f) Repeat the virtual server creation procedure, and then from the **Protocol** list, choose **UDP**. Also choose the SSL client, SSL server, and Authentication profiles from their respective lists as needed.

   You now have a TCP and UDP virtual server to handle SIP traffic.

You now have a SIP ALG profile for use by CGNAT.

Configuring a CGNAT iRule

You create iRules® to automate traffic forwarding for XML content-based routing. When a match occurs, an iRule event is triggered, and the iRule directs the individual request to an LSN pool, a node, or virtual server.

1. On the Main tab, click **Carrier Grade NAT** > **iRules**.
   The iRule List screen opens.
2. Click **Create**.
3. In the **Name** field, type a 1 to 31 character name, such as `cgn_https_redirect_iRule`.
4. In the **Definition** field, type the syntax for the iRule using Tool Command Language (Tcl) syntax.
   For complete and detailed information iRules syntax, see the F5 Networks DevCentral web site (`http://devcentral.f5.com`).
5. Click **Finished**.

You now have an iRule to use with a CGNAT virtual server.

Chapter 3

Using NAT44 to Translate IPv4 Addresses

About NAT44

For the BIG-IP® system CGNAT module, NAT44 is the NAT type that maps IPv4 subscriber private addresses to IPv4 Internet public addresses. Translation addresses and ports are set in LSN pools. The CGNAT module performs NAT44 translations for all IP traffic.

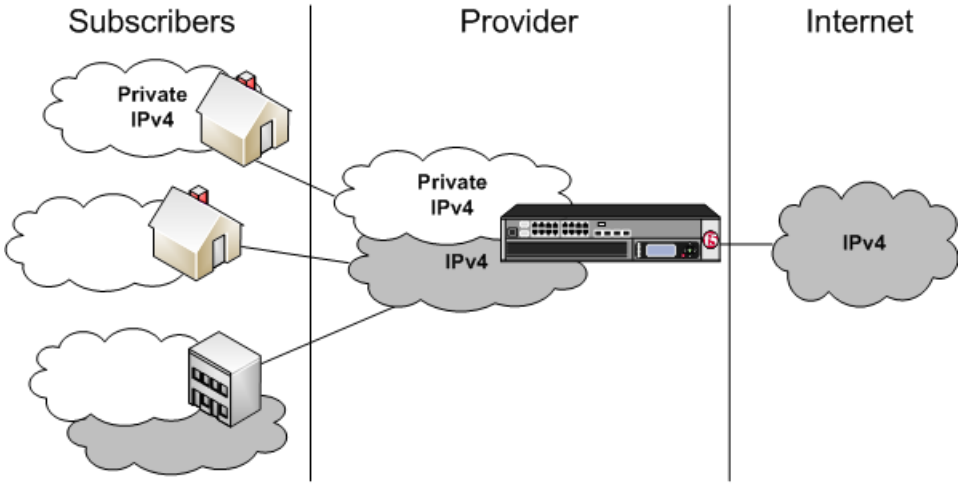

Figure 2: Diagram of a NAT44 network

About CGNAT hairpinning

An optional feature on the BIG-IP system, *hairpinning* routes traffic from one subscriber's client to an external address of another subscriber's server, where both client and server are located in the same subnet. To each subscriber, it appears that the other subscriber's address is on an external host and on a different subnet. The BIG-IP system can recognize this situation and send, or hairpin, the message back to the origin subnet so that the message can reach its destination.

Note: *At present hairpinning works with all BIG-IP CGNAT scenarios except NAT64.*

Task summary

Perform these tasks to translate IPv4 addresses using NAT44.

Creating an LSN pool

The CGNAT module must be enabled through **System** > **Resource Provisioning** before LSN pools can be configured.

Large Scale NAT (LSN) pools are used by the CGNAT module to allow efficient configuration of translation prefixes and parameters.

1. On the Main tab, click **Carrier Grade NAT** > **LSN Pools**.
   The LSN Pool List screen opens.
2. Click **Create**.
3. Enter a unique name in the **Name** field.
4. In the Configuration area, for the **Member List** setting, enter an address and a prefix length in the **Address/Prefix Length** field and click **Add**.

   If your pool uses deterministic mode, ensure that any address ranges you enter as a member do not overlap another member's prefix address ranges. For example, the address and prefix `10.10.10.0/24` overlaps `10.10.10.0/23`.
5. Click **Finished**.

Your LSN pool is now ready and you can continue to configure your CGNAT.

Creating a virtual server for an LSN pool

Virtual servers are matched based on source (client) addresses. Define a virtual server that references the CGNAT profile and the LSN pool.

1. On the Main tab, click **Carrier Grade NAT** > **Virtual Servers**.
   The Virtual Servers screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. From the **Type** list, select **Performance (Layer 4)**.
5. For the **Destination** setting, in the **Address** field, type `0.0.0.0` to allow all traffic to be translated.
6. In the **Service Port** field, type `*` or select *** All Ports** from the list.
7. From the **VLAN and Tunnel Traffic** list, select **Enabled on**. Then, for the **VLANs and Tunnels** setting, move the VLAN or VLANs on which you want to allow the virtual servers to share traffic from the **Available** list to the **Selected** list.
8. For the **LSN Pool** setting, select the pool that this server will draw on for translation addresses.
9. In the Resources area of the screen, for the **iRules** setting, select the name of the iRule that you want to assign and using the Move button, move the name from the **Available** list to the **Enabled** list.
10. Click **Finished**.

The custom CGNAT virtual server now appears in the CGNAT Virtual Servers list.

Configuring a SIP ALG profile

You must have a SIP registrar and proxy configured prior to using a SIP ALG profile.

The SIP ALG profile provides the CGNAT module with enough protocol and service knowledge to make specified packet modifications to the IP and TCP/UDP headers, as well as the SIP payload during translation.

Important: *Only edit copies of the included ALG profiles to avoid unwanted propagation of settings to other profiles that use the included profiles as parents.*

1. On the Main tab, click **Carrier Grade NAT** > **ALG Profiles** > **SIP**.
   The SIP screen opens and displays a list of available SIP ALG profiles.
2. Click **Create** to open the New SIP Profile screen.
3. Enter a name for the new profile.
4. From the **Parent Profile** list, select **sip** as the new profile.
5. For the **Terminate on BYE** setting, select the **Enabled** check box.
6. Select the **Dialog Aware** check box, and enter a unique community string in the **Community** field.
7. From the **Insert Via Header** list, select **Enabled**.
8. Click **Finished** to save the new SIP ALG profile.
9. You must also create two virtual servers: one to handle SIP TCP traffic and another to handle SIP UDP traffic.
   a) Create a host virtual server with a **Source** address of `0.0.0.0/0` and a **Destination** type set as **Network**, as well as a **Mask** of `0.0.0.0` and a **Service Port** of `5060`.
   b) From the **Protocol** list, select **TCP**.
   c) From the **SIP Profile** list, select a SIP profile.
   d) From the **VLAN and Tunnel Traffic** list, select **All VLANs and Tunnels**.
   e) From the **LSN Pool** list, select an LSN pool.
   f) Repeat the virtual server creation procedure, and then from the **Protocol** list, choose **UDP**. Also choose the SSL client, SSL server, and Authentication profiles from their respective lists as needed.

   You now have a TCP and UDP virtual server to handle SIP traffic.

You now have a SIP ALG profile for use by CGNAT.

Configuring a CGNAT iRule

You create iRules® to automate traffic forwarding for XML content-based routing. When a match occurs, an iRule event is triggered, and the iRule directs the individual request to an LSN pool, a node, or virtual server.

1. On the Main tab, click **Carrier Grade NAT** > **iRules**.
   The iRule List screen opens.
2. Click **Create**.
3. In the **Name** field, type a 1 to 31 character name, such as `cgn_https_redirect_iRule`.
4. In the **Definition** field, type the syntax for the iRule using Tool Command Language (Tcl) syntax.
   For complete and detailed information iRules syntax, see the F5 Networks DevCentral web site (`http://devcentral.f5.com`).
5. Click **Finished**.

You now have an iRule to use with a CGNAT virtual server.

Chapter 4

Using Deterministic Mode to Simplify Logging

About deterministic address translation mode

Deterministic address translation mode provides address translation that eliminates logging of every address mapping, while still allowing internal client address tracking using only an external address and port, and a destination address and port. Reverse mapping allows BIG-IP® CGNAT operators to respond to legal requests revealing the identity of the originator of a specific communication. A typical example is revealing the identity of file sharers or P2P network users accused of copyright theft.

Deterministic mode allows unique identification of internal client address based on:

- External address and port (the address and port visible to the destination server)
- Destination address and port (the service accessed by the client)
- Time

Restrictions

Deterministic mode has the configuration restrictions listed here:

- Only NAT44 can use deterministic mode.
- The subscriber (client-side) and Internet (server-side) interfaces (VLANs) must be set either as a source or destination address in the CMP Hash setting.
- The complete set of all internal client addresses that will ever communicate through the CGNAT must be entered at configuration time.

Note: *This means that all virtual servers referring to an LSN pool through deterministic NAT mode must specify the source attribute with a value other than 0.0.0.0/0 or ::/0 (any/0, any6/0).*

- Use only the most specific address prefixes covering all customer addresses.
- Members of two or more deterministic LSN pools must not overlap; in other words, every external address used for deterministic mapping must occur only in one LSN pool.

Simplified logging

As an alternative to per-connection logging, deterministic mode maps internal addresses to external addresses algorithmically to calculate the mapping without relying on per-connection logging. Deterministic mode significantly reduces the logging burden while mapping a subscriber's inside IP address with an outside Internet address and port.

To decipher mapping generated by LSN pools using deterministic mode, you must use the DNAT utility that can be run from the system's tmsh command prompt.

Task summary

Perform these tasks to use Deterministic mode for logging.

Creating a deterministic LSN pool
Creating a VLAN for a deterministic NAT
Creating a virtual server for an LSN pool

Creating a deterministic LSN pool

The CGNAT module must be provisioned before LSN pools can be configured.

Large Scale NAT (LSN) pools are used by the CGNAT module to allow efficient configuration of translation prefixes and parameters.

1. On the Main tab, click **Carrier Grade NAT** > **LSN Pools**.
   The LSN Pool List screen opens.
2. Click **Create**.
3. Enter a unique name in the **Name** field.
4. Select **Deterministic** for the pool's translation **Mode**.
   Note that deterministic mode does not support *DS-lite* tunneling or *NAT64*.
5. In the Configuration area, for the **Member List** setting, enter an address and a prefix length in the **Address/Prefix Length** field and click **Add**.
   If your pool uses deterministic mode, ensure that any address ranges you enter as a member do not overlap another member's prefix address ranges. For example, the address and prefix `10.10.10.0/24` overlaps `10.10.10.0/23`.
6. For deterministic mode, the **Backup Member List** must have at least one member so enter an address in the **Address/Prefix Length** field and click **Add**.
7. Click **Finished**.

Your deterministic LSN pool is now ready and you can continue to configure your CGNAT.

Creating a VLAN for a deterministic NAT

VLANs represent a collection of hosts that can share network resources, regardless of their physical location on the network. You create a VLAN to associate physical interfaces with that VLAN.

1. On the Main tab, click **Network** > **VLANs**.
   The VLAN List screen opens.
2. Click **Create**.
   The New VLAN screen opens.
3. In the **Name** field, type a unique name for the VLAN.
4. In the **Tag** field, type a numeric tag, from 1 - 4094 , for the VLAN, or leave the field blank if you want the BIG-IP system to automatically assign a VLAN tag.
   The VLAN tag identifies the traffic from hosts in the associated VLAN.
5. For the **Interfaces** setting, from the **Available** list, click an interface number or trunk name and add the selected interface or trunk to the **Untagged** list. Repeat this step as necessary.
6. From the **Configuration** list, select **Advanced**.
7. Select the **Source Check** check box if you want the system to verify that the return route to an initial packet is the same VLAN from which the packet originated.
8. In the **MTU** field, retain the default number of bytes (**1500**).
9. Click **Finished**.
   The screen refreshes, and displays the new VLAN in the list.
10. Click the name of the newly created VLAN to open it.
11. From the **Configuration** list, select **Advanced**.

12. From the **CMP Hash** list, select **Source** if this VLAN is the subscriber side or **Destination** if this VLAN is the Internet side.
13. Click **Finished**.
    The screen refreshes, and displays the new VLAN in the list.

You now have one of two VLANs for your deterministic NAT. Repeat these steps to create a second VLAN to act as the destination if the first VLAN is the source or vice versa.

Creating a virtual server for an LSN pool

Virtual servers are matched based on source (client) addresses. Define a virtual server that references the CGNAT profile and the LSN pool.

1. On the Main tab, click **Carrier Grade NAT** > **Virtual Servers**.
   The Virtual Servers screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. From the **Type** list, select **Performance (Layer 4)**.
5. For the **Destination** setting, in the **Address** field, type `0.0.0.0` to allow all traffic to be translated.
6. In the **Service Port** field, type `*` or select *** All Ports** from the list.
7. From the **VLAN and Tunnel Traffic** list, select **Enabled on**. Then, for the **VLANs and Tunnels** setting, move the VLAN or VLANs on which you want to allow the virtual servers to share traffic from the **Available** list to the **Selected** list.
8. For the **LSN Pool** setting, select the pool that this server will draw on for translation addresses.
9. In the Resources area of the screen, for the **iRules** setting, select the name of the iRule that you want to assign and using the Move button, move the name from the **Available** list to the **Enabled** list.
10. Click **Finished**.

The custom CGNAT virtual server now appears in the CGNAT Virtual Servers list.

Chapter 5

Configuring Local CGNAT Logging

Overview: Configuring local logging for CGNAT

You can configure the BIG-IP® system to send log messages about carrier grade network address translation (CGNAT) processes to the local Syslog database on the BIG-IP system.

Note: *Enabling logging impacts BIG-IP system performance.*

When configuring local logging of CGNAT processes, it is helpful to understand the objects you need to create and why:

Object to create in implementation	Reason
Destination (formatted/local)	Create a formatted log destination to format the logs in human-readable name/value pairs, and forward the logs to the local-syslog database.
Publisher (local-syslog)	Create a log publisher to send logs to the previously created destination that formats the logs in name/value pairs, and forwards the logs to the local Syslog database on the BIG-IP system.
LSN pool	Associate a large scale NAT (LSN) pool with a log publisher in order to log messages about the traffic handled by the pool.

Task summary

Perform these tasks to configure local logging of CGNAT processes on the BIG-IP® system.

Creating a formatted local log destination for CGNAT
Creating a publisher to send log messages to the local Syslog database
Configuring an LSN pool with a local Syslog log publisher

Creating a formatted local log destination for CGNAT

Create a formatted logging destination to specify that log messages about CGNAT processes are sent to the local Syslog database in a format that displays name/value pairs in a human-readable format.

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Destinations**.
   The Log Destinations screen opens.
2. Click **Create**.
3. In the **Name** field, type a unique, identifiable name for this destination.
4. From the **Type** list, select **Splunk**.
5. From the **Forward To** list, select **local-syslog**.
6. Click **Finished**.

Creating a publisher to send log messages to the local Syslog database

Create a publisher to specify that the BIG-IP® system sends formatted log messages to the local Syslog database, on the BIG-IP system.

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Publishers**.
   The Log Publishers screen opens.
2. Click **Create**.
3. In the **Name** field, type a unique, identifiable name for this publisher.
4. For the **Destinations** setting, select the previously created destination from the **Available** list, which formats the logs in the Splunk format and forwards the logs to the local Syslog database; click **<<** to move the destination to the **Selected** list.
5. Click **Finished**.

Configuring an LSN pool with a local Syslog log publisher

Before associating a large scale NAT (LSN) pool with a log publisher, ensure that at least one log publisher exists that sends formatted log messages to the local Syslog database on the BIG-IP® system.

Associate an LSN pool with the log publisher that the BIG-IP system uses to send formatted log messages to the local Syslog database.

1. On the Main tab, click **Carrier Grade NAT** > **LSN Pools**.
   The LSN Pool List screen opens.
2. Select an LSN pool from the list.
3. From the **Log Publisher** list, select the log publisher that sends formatted log messages to the local Syslog database on the BIG-IP system.
4. Click **Finished**.

Implementation result

You now have an implementation in which the BIG-IP® system logs messages about CGNAT processes and sends the log messages to the local Syslog database on the BIG-IP system.

Chapter 6

Configuring High-Speed Remote CGNAT Logging

Overview: Configuring remote high-speed logging for CGNAT

You can configure the BIG-IP® system to log information about carrier grade network address translation (CGNAT) processes and send the log messages to remote high-speed log servers.

When configuring remote high-speed logging of CGNAT processes, it is helpful to understand the objects you need to create and why, as described here:

Object to create in implementation	Reason
Pool of remote log servers	Create a pool of remote log servers to which the BIG-IP system can send log messages.
Destination (unformatted)	Create a log destination of Remote High-Speed Log type that specifies a pool of remote log servers.
Destination (formatted)	If your remote log servers are the ArcSight, Splunk, or Remote Syslog type, create an additional log destination to format the logs in the required format and forward the logs to a remote high-speed log destination.
Publisher	Create a log publisher to send logs to a set of specified log destinations.
LSN pool	Associate a large scale NAT (LSN) pool with a log publisher in order to log messages about the traffic handled by the pool.

This illustration shows the association of the configuration objects for remote high-speed logging of CGNAT processes.

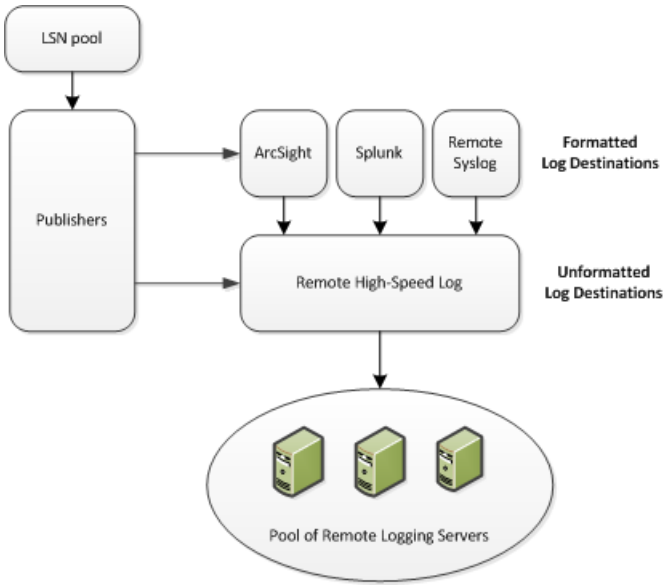

Figure 3: Association of remote high-speed logging configuration objects

Task summary

Perform these tasks to configure remote high-speed logging of CGNAT processes on the BIG-IP® system.

Note: *Enabling remote high-speed logging impacts BIG-IP system performance.*

Creating a pool of remote logging servers

Before creating a pool of log servers, gather the IP addresses of the servers that you want to include in the pool. Ensure that the remote log servers are configured to listen to and receive log messages from the BIG-IP® system.

Create a pool of remote log servers to which the BIG-IP system can send log messages.

1. On the Main tab, click **Local Traffic** > **Pools**.
   The Pool List screen opens.
2. Click **Create**.
   The New Pool screen opens.
3. In the **Name** field, type a unique name for the pool.
4. Using the **New Members** setting, add the IP address for each remote logging server that you want to include in the pool:
   a) Type an IP address in the **Address** field, or select a node address from the **Node List**.
   b) Type a service number in the **Service Port** field, or select a service name from the list.

      Note: *Typical remote logging servers require port* `514`.

   c) Click **Add**.
5. Click **Finished**.

Creating a remote high-speed log destination

Before creating a remote high-speed log destination, ensure that at least one pool of remote log servers exists on the BIG-IP® system.

Create a log destination of the **Remote High-Speed Log** type to specify that log messages are sent to a pool of remote log servers.

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Destinations**.
   The Log Destinations screen opens.
2. Click **Create**.
3. In the **Name** field, type a unique, identifiable name for this destination.
4. From the **Type** list, select **Remote High-Speed Log**.

Important: *If you use log servers such as Remote Syslog, Splunk, or ArcSight, which require data be sent to the servers in a specific format, you must create an additional log destination of the required type, and associate it with a log destination of the* ***Remote High-Speed Log*** *type. This allows the BIG-IP system to send data to the servers in the required format.*

The BIG-IP system is configured to send an unformatted string of text to the log servers.

5. From the **Pool Name** list, select the pool of remote log servers to which you want the BIG-IP system to send log messages.
6. From the **Protocol** list, select the protocol used by the high-speed logging pool members.
7. Click **Finished**.

Creating a formatted remote high-speed log destination

Ensure that at least one remote high-speed log destination exists on the BIG-IP® system.

Create a formatted logging destination to specify that log messages are sent to a pool of remote log servers, such as Remote Syslog, Splunk, or ArcSight servers.

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Destinations**.
   The Log Destinations screen opens.
2. Click **Create**.
3. In the **Name** field, type a unique, identifiable name for this destination.
4. From the **Type** list, select a formatted logging destination, such as **Remote Syslog**, **Splunk**, or **ArcSight**.

Important: *ArcSight formatting is only available for logs coming from the network Application Firewall Manager (AFM) and the Application Security Manager (ASM™).*

The BIG-IP system is configured to send a formatted string of text to the log servers.

5. If you selected **Remote Syslog**, from the **Syslog Format** list, select a format for the logs, and then from the **High-Speed Log Destination** list, select the destination that points to a pool of remote Syslog servers to which you want the BIG-IP system to send log messages.
6. If you selected **Splunk** or **ArcSight**, from the **Forward To** list, select the destination that points to a pool of high-speed log servers to which you want the BIG-IP system to send log messages.
7. Click **Finished**.

Creating a publisher

Ensure that at least one destination associated with a pool of remote log servers exists on the BIG-IP® system.

Create a publisher to specify where the BIG-IP system sends log messages for specific resources.

1. On the Main tab, click **System** > **Logs** > **Configuration** > **Log Publishers**.
   The Log Publishers screen opens.
2. Click **Create**.
3. In the **Name** field, type a unique, identifiable name for this publisher.
4. For the **Destinations** setting, select a destination from the **Available** list, and click **<<** to move the destination to the **Selected** list.

Note: *If you are using a formatted destination, select the destination that matches your log servers, such as Remote Syslog, Splunk, or ArcSight.*

---

5. Click **Finished**.

Configuring an LSN pool with a log publisher

Before associating a large scale NAT (LSN) pool with a log publisher, ensure that at least one log publisher exists on the BIG-IP® system.

Associate an LSN pool with a log publisher that the BIG-IP system uses to send log messages to a specified destination.

1. On the Main tab, click **Carrier Grade NAT** > **LSN Pools**.
   The LSN Pool List screen opens.
2. Select an LSN pool from the list.
3. From the **Log Publisher** list, select the log publisher the BIG-IP system uses to send log messages to a specified destination.
4. Click **Finished**.

After performing this task, you have an LSN pool for which the BIG-IP system logs messages using the specified log publisher.

Implementation result

Now you have an implementation in which the BIG-IP® system logs messages about CGNAT processes and sends the log messages to a pool of remote log servers.

Chapter 7

Using the Deterministic NAT log tool

About the DNAT utility

The deterministic NAT (DNAT) utility allows calculation of forward and reverse source address and port mapping of deterministic-mode LSN pools, by using the states stored in the analyzed TMM log file.

Using the DNAT utility to lookup deterministic NAT mappings

A knowledge of navigating in tmsh is suggested before using the DNAT utility. For detailed information about navigating in tmsh, see the *Traffic Management Shell (tmsh) Reference Guide*.

Deterministic NATs can reduce total log file size but require use of the DNAT utility (available in tmsh) to decipher the mapping. With the DNAT utility, you can calculate forward and reverse source address and port mapping of an LSN pool using deterministic mode based on the state stored in the specified TMM log file.

1. Use an SSH tool to access the BIG-IP® system from the command line.
2. At the command line, type: `tmsh`.
   This starts `tmsh` in interactive shell mode and displays the prompt: `(tmos)#`.
3. To show a list of translation address/port pairs used for a subscriber at 10.0.0.1:4321 connecting to 65.61.115.222:80, using the deterministic NAT states contained in `/var/log/ltm`, type the command: `run util dnat --file /var/log/ltm --client_addr 10.0.0.1 --client_port 4321 --server_addr 65.61.115.222 --action forward`

   Replace these example addresses with your actual client and server.

   This displays a list of the address/port pairs.
4. To calculate a reverse mapping back to the subscriber address for the connection between 173.240.102.139:5678 and 65.61.115.222:80, using the DNAT states contained in `/var/log/ltm.1`, type the command: `run util dnat --file /var/log/ltm.1 --server_addr 65.61.115.222 --client_addr 173.240.102.139 --client_port 5678 --action reverse`
   This displays the reverse mapping.
5. For more information about the DNAT utility, type the command: `help util dnat` at the `tmsh` prompt.
   The help file for the DNAT utility is displayed.

You now have the basic details for deciphering deterministic log files using the DNAT utility in tmsh.

Chapter 8

Using DS-Lite with CGNAT

Overview: DS-Lite Configuration on BIG-IP systems

As IPv4 addresses are becoming depleted, service providers (DSL, cable, and mobile) face the challenge of supplying IP addresses to new customers. Providing IPv6 addresses alone is often not workable, because most of the public Internet still uses only IPv4, and many customer systems do not yet fully support IPv6. The Dual-Stack Lite (DS-Lite) tunneling technology is one solution to this problem. DS-Lite gives service providers the means to migrate to an IPv6 access network without changing end user devices or software.

What is DS-Lite?

DS-Lite is an IPv4-to-IPv6 transition technology, described in RFC 6333, that uses tunneling and network address translation (NAT) to send IPv4 packets over an IPv6 network. This technology makes it possible, for example, for a service provider with an IPv6 backbone to properly route traffic while overlapping IPv4 networks.

How does DS-Lite work?

The customer-premises equipment (CPE), known as the B4 (Basic Bridging BroadBand) device, encapsulates the IPv4 packets inside IPv6 packets, and sends them to the AFTR (Address Family Transition Router) device. The AFTR device includes carrier-grade NAT (CGNAT), which has a global IPv4 address space. The AFTR device decapsulates the IPv4 traffic and performs address translation, as it sends the traffic to the external IPv4 network.

How does F5 implement DS-Lite?

On the BIG-IP® system, a DS-Lite tunnel is a variation of IPIP tunnels that uses augmented flow lookups to route traffic. *Augmented flow lookups* include the IPv6 address of the tunnel to identify the accurate source of packets that might have the same IPv4 address. When the BIG-IP device receives an IPv6 encapsulated packet, the system terminates the tunnel, decapsulates the packet, and marks it for DS-Lite. When the system re-injects the packet into the IP stack, it performs an augmented flow lookup to properly route the response.

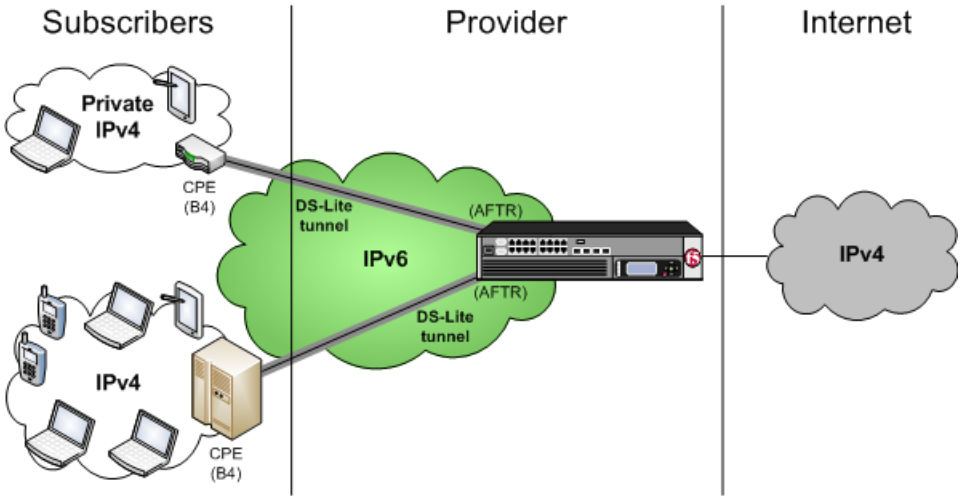

Figure 4: Example of a DS-Lite configuration

About CGNAT hairpinning

An optional feature on the BIG-IP system, *hairpinning* routes traffic from one subscriber's client to an external address of another subscriber's server, where both client and server are located in the same subnet. To each subscriber, it appears that the other subscriber's address is on an external host and on a different subnet. The BIG-IP system can recognize this situation and send, or hairpin, the message back to the origin subnet so that the message can reach its destination.

Note: *At present hairpinning works with all BIG-IP CGNAT scenarios except NAT64.*

Task summary

When you set up DS-Lite, you must configure devices at both ends of the tunnel: the B4 device and the AFTR device. For this implementation, the AFTR device is a BIG-IP® system.

Before you configure the AFTR device, set up your CPE as a B4 device, and configure it to send traffic to the v6 self IP address of the BIG-IP® system. For instructions, consult the manufacturer's documentation for your device.

Creating a DS-Lite tunnel on the BIG-IP as an AFTR device
Assigning a self IP address to an AFTR device
Configuring CGNAT for DS-Lite

Verifying traffic statistics for a DS-Lite tunnel

Creating a DS-Lite tunnel on the BIG-IP as an AFTR device

Before you configure the tunnel, ensure that the BIG-IP® device you are configuring has an IPv6 address.

You can create a DS-Lite (wildcard) tunnel for terminating IPv4-in-IPv6 tunnels to remote B4 devices, and recycling the IPv4 address space.

1. On the Main tab, click **Network** > **Tunnels** > **Tunnel List** > **Create**.
   The New Tunnel screen opens.
2. In the **Name** field, type a unique name for the tunnel.
3. From the **Encapsulation Type** list, select **dslite**.
4. In the **Local Address** field, type the IPv6 address of the local BIG-IP device.
5. From the **Remote Address** list, select **Specify**, and type `::`.
   This value must be a wildcard IP address.
6. Click **Finished**.

You have now created a DS-Lite tunnel that functions as an AFTR (Address Family Translation Router) device.

Assigning a self IP address to an AFTR device

Ensure that you have created a DS-Lite tunnel.

Self IP addresses can enable the BIG-IP® system, and other devices on the network, to route application traffic through the associated tunnel.

1. On the Main tab, click **Network** > **Self IPs**.
   The Self IPs screen opens.
2. Click **Create**.
   The New Self IP screen opens.
3. In the **Name** field, type a unique name for the self IP.
4. In the **IP Address** field, type an IP address.
   This IP address is the IPv4 gateway that the B4 devices use to reach the Internet. F5 recommends using the IP address space that the IANA has specifically allocated for an AFTR device, for example, `192.0.0.1`.
5. In the **Netmask** field, type the network mask for the specified IP address.
6. From the **VLAN/Tunnel** list, select the tunnel with which to associate this self IP address.
7. Click **Finished**.

Configuring CGNAT for DS-Lite

Before starting this task, ensure that CGNAT is licensed and the feature module enabled on the BIG-IP® system, and you have created at least one LSN pool.

When you are configuring DS-Lite, you must set up a forwarding virtual server to provide the Large Scale NAT (LSN), which is specified by the DS-Lite tunnel as an augmented flow lookup.

1. On the Main tab, click **Carrier Grade NAT** > **Virtual Servers**.
   The Virtual Servers screen opens.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. In the **Name** field, type a unique name for the virtual server.
4. From the **Type** list, select **Performance (Layer 4)**.
5. For the **Destination** setting, select **Network**, and type `0.0.0.0` in the **Address** field and `0.0.0.0` in the **Mask** field.
6. In the **Service Port** field, type `*` or select *** All Ports** from the list.
7. From the **Configuration** list, select **Advanced**.
8. From the **Protocol** list, select *** All Protocols**.
9. From the **LSN Pool** list, select an LSN pool.
10. Click **Finished**.

This virtual server now intercepts traffic leaving the DS-Lite tunnel, provides the LSN address translation, and forwards the traffic to the IPv4 gateway.

Verifying traffic statistics for a DS-Lite tunnel

After you configure DS-Lite on a BIG-IP® system, you can check the statistics for the tunnel to verify that traffic is passing through it.

1. Log on to the BIG-IP command-line interface.
2. At the command prompt, type `tmsh show sys connection all-properties`.
   The result should show tunnel with `any` as the remote endpoint (on the first line), and `ipencap` as the `Protocol`, as shown in the example.

```
2001:db8::/32.any - 2001:db8::46.any - any6.any - any6.any
-------------------------------------------------------------
  TMM           0
  Type          any
  Acceleration  none
  Protocol      ipencap
  Idle Time     1
  Idle Timeout  300
  Unit ID       1
  Lasthop       /Common/wan 00:d0:01:b9:88:00
  Virtual Path  2001:db8::46.any

                      ClientSide  ServerSide
  Client Addr  2001:db8::45.any    any6.any
  Server Addr  2001:db8::46.any    any6.any
  Bits In                171.6K           0
  Bits Out               171.6K           0
```

CGNAT Glossary

carrier-grade network address translation (CGNAT)
A scalable and flexible type of network address translation available to service providers and enterprise BIG-IP® users for subscriber outbound source traffic.

Customer premise equipment (CPE)
Customer premise equipment includes devices such as cable modems, DSL routers, smart phones, and any other device at the subscriber's premises that directly connects to the service provider.

deterministic NAT
A network address translation that maintains lawful logging (per *RFC 6269*), while avoiding the need to dynamically log all NAT mappings so that a service provider can substantially reduce the number of subscriber records that must be retained. This translation is also known as port block allocation.

large-scale NAT (LSN) pool
A large-scale NAT pool or LSN pool intended for configurations with many subscribers and only available to the CGNAT module, such as subscriber outbound source NAT, when many subscribers want to access Internet services. It is equally applicable to service provider and enterprise organizations.

NAT44
Translates internal IPv4 traffic to Internet IPv4 destinations.

NAT64
Translates internal IPv6 traffic to the Internet IPv4 destinations.

port block allocation
A port block allocation (PBA) is another term used for a deterministic NAT.

subscriber
A service provider's customer.

tunnel
TCP and UDP traffic, where the payload is other IP traffic routed through a tunneling protocol to carry payloads over an incompatible delivery-network, or provide secure transport through an untrusted network.

Index