BIG-IP® System: Clustering with ECMP

Version 11.6

Table of Contents

Legal Notices....5
Acknowledgments....7

Chapter 1: Introduction to All-Active Clustering using ECMP....19
Overview: Configuring all-active clustering using ECMP....20

Chapter 2: Configuring Basic BIG-IP System Settings....21
Overview: Configuring basic system settings....22
Confirming the contents of the BIG-IP license....22
Viewing the DNS server configuration....22
Specifying a list of NTP servers....23
Creating VLANs....23
Creating self IP addresses....24
Enabling dynamic routing protocols for route domain 0....26
Specifying an IP address for config sync....26
Establishing device trust....27
Creating a Sync-Only device group....28
Syncing the BIG-IP configuration to the device group....29
Creating an administrative partition....29
Changing the current partition....30

Chapter 3: Use Case 1: Creating a Configuration that Uses SNAT Auto Map....31
Using SNAT Auto Map....32
Creating a load balancing pool....32
Creating a virtual server....33
Confirming virtual address exclusion from a traffic group....34
Syncing the BIG-IP configuration to the device group....35
Configuring the BGP protocol....35
Implementation result....36

Chapter 4: Use Case 2: Creating a Configuration that Uses a SNAT Pool....39
Using a SNAT pool....40
Creating a load balancing pool....41
Defining a route to the server....42
Creating SNAT pools....42
Creating a string data group....43
Creating an iRule for SNAT pool selection....43
Creating a virtual server....44
Confirming virtual address exclusion from a traffic group....45

Syncing the BIG-IP configuration to the device group.....45
Configuring the BGP protocol.....46
Implementation result.....47

Legal Notices

Publication Date

This document was published on November 12, 2014.

Publication Number

MAN-0545-00

Copyright

Trademarks

Patents

This product may be protected by one or more patents indicated at:
http://www.f5.com/about/guidelines-policies/patents

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 software with glib library utility functions, which is protected under the GNU Public License.

This product includes software with grub2 bootloader functions, which is protected under the GNU Public License.

This product includes software with the Intel Gigabit Linux driver, which is protected under the GNU Public License. Copyright ©1999 - 2012 Intel Corporation.

This product includes software with the Intel 10 Gigabit PCI Express Linux driver, which is protected under the GNU Public License. Copyright ©1999 - 2012 Intel Corporation.

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 software developed by Andrew Tridgell, which is protected under the GNU Public License, copyright ©1992-2000.

This product includes software developed by Jeremy Allison, which is protected under the GNU Public License, copyright ©1998.

This product includes software developed by Guenther Deschner, which is protected under the GNU Public License, copyright ©2008.

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

This product includes gd-libgd library software developed by the following in accordance with the following copyrights:

- Portions copyright ©1994, 1995, 1996, 1997, 1998, 2000, 2001, 2002 by Cold Spring Harbor Laboratory. Funded under Grant P41-RR02188 by the National Institutes of Health.
- Portions copyright ©1996, 1997, 1998, 1999, 2000, 2001, 2002 by Boutell.Com, Inc.
- Portions relating to GD2 format copyright ©1999, 2000, 2001, 2002 Philip Warner.
- Portions relating to PNG copyright ©1999, 2000, 2001, 2002 Greg Roelofs.
- Portions relating to gdttf.c copyright ©1999, 2000, 2001, 2002 John Ellson (ellson@lucent.com).
- Portions relating to gdft.c copyright ©2001, 2002 John Ellson (ellson@lucent.com).
- Portions copyright ©2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007 2008 Pierre-Alain Joye (pierre@libgd.org).
- Portions relating to JPEG and to color quantization copyright ©2000, 2001, 2002, Doug Becker and copyright ©1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, Thomas G. Lane. This software is based in part on the work of the Independent JPEG Group.
- Portions relating to WBMP copyright 2000, 2001, 2002 Maurice Szmurlo and Johan Van den Brande. Permission has been granted to copy, distribute and modify gd in any context without fee, including a commercial application, provided that this notice is present in user-accessible supporting documentation.

This product includes software developed by Oracle America, Inc. Copyright ©2012.

1. Java Technology Restrictions. Licensee shall not create, modify, change the behavior of, or authorize licensees of licensee to create, modify, or change the behavior of, classes, interfaces, or subpackages that are in any way identified as "java", "javax", "sun" or similar convention as specified by Oracle in any naming convention designation. In the event that Licensee creates an additional API(s) which: (a) extends the functionality of a Java Environment; and (b) is exposed to third party software developers for the purpose of developing additional software which invokes such additional API, Licensee must promptly publish broadly an accurate specification for such API for free use by all developer.
2. Trademarks and Logos. This License does not authorize an end user licensee to use any Oracle America, Inc. name, trademark, service mark, logo or icon. The end user licensee acknowledges that Oracle owns the Java trademark and all Java-related trademarks, logos and icon including the Coffee Cup and Duke ("Java Marks") and agrees to: (a) comply with the Java Trademark Guidelines at http://www.oraclc.com/html/3party.html; (b) not do anything harmful to or inconsistent with Oracle's rights in the Java Marks; and (c) assist Oracle in protecting those rights, including assigning to Oracle any rights acquired by Licensee in any Java Mark.
3. Source Code. Software may contain source code that, unless expressly licensed for other purposes, is provided solely for reference purposes pursuant to the terms of your license. Source code may not be redistributed unless expressly provided for in the terms of your license.
4. Third Party Code. Additional copyright notices and license terms applicable to portion of the Software are set forth in the THIRDPARTYLICENSEREADME.txt file.
5. Commercial Features. Use of the Commercial Features for any commercial or production purpose requires a separate license from Oracle. "Commercial Features" means those features identified in Table I-I (Commercial Features In Java SE Product Editions) of tile Software documentation accessible at http://www.oracle.com/technetwork/java/javase/documentation/index.html.

This product includes utilities developed by Linus Torvalds for inspecting devices connected to a USB bus.

This product includes perl-PHP-Serialization software, developed by Jesse Brown, copyright ©2003, and distributed under the Perl Development Artistic License (http://dev.perl.org/licenses/artistic.html).

This product includes software developed by members of the CentOS Project under the GNU Public License, copyright ©2004-2011 by the CentOS Project.

OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This product includes software developed by Douglas Gilbert. Copyright ©1992 - 2012 The FreeBSD Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE FREEBSD PROJECT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FREEBSD PROJECT OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project.

This product includes software developed as open source software. Copyright ©1994 - 2012 The FreeBSD Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). Copyright ©1998 - 2011 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes software licensed from William Ferrell, Selene Scriven and many other contributors under the GNU General Public License, copyright ©1998 - 2006.

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 includes software developed by the Computer Systems Engineering Group at Lawrence Berkeley Laboratory. Copyright ©1990-1994 Regents of the University of California. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: This product includes software developed by the Computer Systems Engineering Group at Lawrence Berkeley Laboratory.

4. Neither the name of the University nor of the Laboratory may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes software developed by Sony Computer Science Laboratories Inc. Copyright © 1997-2003 Sony Computer Science Laboratories Inc. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY SONY CSL AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL SONY CSL OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

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.

This product includes software developed by Jeremy Ashkenas and DocumentCloud, and distributed under the MIT license. Copyright © 2010-2013 Jeremy Ashkenas, DocumentCloud.

This product includes gson software, distributed under the Apache License version 2.0. Copyright © 2008-2011 Google Inc.

This product includes owasp-jave-encoder software, copyright © 2014, Jeff Ichnowski, and distributed under the New BSD license.

This product may include Intel SDD software subject to the following license; check your hardware specification for details.

1. LICENSE. This Software is licensed for use only in conjunction with Intel solid state drive (SSD) products. Use of the Software in conjunction with non-Intel SSD products is not licensed hereunder. Subject to the terms of this Agreement, Intel grants to You a nonexclusive, nontransferable, worldwide, fully paid-up license under Intel's copyrights to:

   - copy the Software onto a single computer or multiple computers for Your personal, noncommercial use; and
   - make appropriate back-up copies of the Software, for use in accordance with Section 1a) above.

   The Software may contain the software or other property of third party suppliers, some of which may be identified in, and licensed in accordance with, any enclosed "license.txt" file or other text or file.

   Except as expressly stated in this Agreement, no license or right is granted to You directly or by implication, inducement, estoppel or otherwise. Intel will have the right to inspect or have an independent auditor inspect Your relevant records to verify Your compliance with the terms and conditions of this Agreement.

2. RESTRICTIONS. You will not:

   a. copy, modify, rent, sell, distribute or transfer any part of the Software, and You agree to prevent unauthorized copying of the Software; and,
   b. reverse engineer, decompile, or disassemble the Software; and,
   c. sublicense or permit simultaneous use of the Software by more than one user; and,
   d. otherwise assign, sublicense, lease, or in any other way transfer or disclose Software to any third party, except as set forth herein; and,
   e. subject the Software, in whole or in part, to any license obligations of Open Source Software including without limitation combining or distributing the Software with Open Source Software in a manner that subjects the Software or any portion of the Software provided by Intel hereunder to any license obligations of such Open Source Software. "Open Source Software" means any software that requires as a condition of use, modification and/or distribution of such software that such software or other software incorporated into, derived from or distributed with such software:

      a. be disclosed or distributed in source code form; or
      b. be licensed by the user to third parties for the purpose of making and/or distributing derivative works; or
      c. be redistributable at no charge.

   Open Source Software includes, without limitation, software licensed or distributed under any of the following licenses or distribution models, or licenses or distribution models substantially similar to any of the following:

   a. GNU’s General Public License (GPL) or Lesser/Library GPL (LGPL),
   b. the Artistic License (e.g., PERL),
   c. the Mozilla Public License,
   d. the Netscape Public License,
   e. the Sun Community Source License (SCSL),
   f. vi) the Sun Industry Source License (SISL),
   g. vii) the Apache Software license, and
   h. viii) the Common Public License (CPL).

3. OWNERSHIP OF SOFTWARE AND COPYRIGHTS. Title to all copies of the Software remains with Intel or its suppliers. The Software is copyrighted and protected by the laws of the United States and other countries, and international treaty provisions. You may not remove any copyright notices from the Software. Intel may make changes to the Software, or to materials referenced therein, at any time and without notice, but is not obligated to support or update the Software. Except as otherwise expressly provided, Intel grants no express or implied right or license under Intel patents, copyrights, trademarks, or other intellectual property rights.
4. Entire Agreement. This Agreement contains the complete and exclusive statement of the agreement between You and Intel and supersedes all proposals, oral or written, and all other communications relating to the subject matter of this Agreement. Only a written instrument duly executed by authorized representatives of Intel and You may modify this Agreement.
5. LIMITED MEDIA WARRANTY. If the Software has been delivered by Intel on physical media, Intel warrants the media to be free from material physical defects for a period of ninety (90) days after delivery by Intel. If such a defect is found, return the media to Intel for replacement or alternate delivery of the Software as Intel may select.
6. EXCLUSION OF OTHER WARRANTIES. EXCEPT AS PROVIDED ABOVE, THE SOFTWARE IS PROVIDED "AS IS" WITHOUT ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND, INCLUDING WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE. Intel does not warrant or assume responsibility for any errors, the accuracy or completeness of any information, text, graphics, links or other materials contained within the Software.
7. LIMITATION OF LIABILITY. IN NO EVENT WILL INTEL OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION OR LOST INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE, EVEN IF INTEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME JURISDICTIONS PROHIBIT EXCLUSION OR LIMITATION OF LIABILITY FOR IMPLIED WARRANTIES OR CONSEQUENTIAL OR INCIDENTAL DAMAGES, SO THE ABOVE LIMITATION MAY NOT APPLY TO YOU. YOU MAY ALSO HAVE OTHER LEGAL RIGHTS THAT VARY FROM JURISDICTION TO JURISDICTION.
8. TERMINATION OF THIS AGREEMENT. Intel may terminate this Agreement at any time if You violate its terms. Upon termination, You will immediately destroy the Software or return all copies of the Software to Intel.
9. APPLICABLE LAWS. Claims arising under this Agreement will be governed by the laws of Delaware, excluding its principles of conflict of laws and the United Nations Convention on Contracts for the Sale of Goods. You may not export the Software in violation of applicable export laws and regulations. Intel is not obligated under any other agreements unless they are in writing and signed by an authorized representative of Intel.
10. GOVERNMENT RESTRICTED RIGHTS. The Software is provided with "RESTRICTED RIGHTS." Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or their successors. Use of the Software by the Government constitutes acknowledgment of Intel's proprietary rights therein. Contractor or Manufacturer is Intel Corporation, 2200 Mission College Blvd., Santa Clara, CA 95054.

Chapter 1

Introduction to All-Active Clustering using ECMP

- *Overview: Configuring all-active clustering using ECMP*

Overview: Configuring all-active clustering using ECMP

You can implement the clustering of BIG-IP® devices on a network through the use of the Equal Cost Multi-Path (ECMP) protocol. Specifically, you can enable ECMP on an upstream router and then configure the same virtual IP address on each BIG-IP device in a Sync-Only device group. In this case, the upstream router uses ECMP to determine, through route advertisement, whether there are multiple, equal-cost paths to the virtual address. If so, the router employs an algorithm to select a path to the virtual address on any one of the BIG-IP devices.

With this configuration, no single application is pinned to a specific device, and no device operates in a standby state. Instead, all devices in the cluster can be active for that application traffic. Moreover, you can add capacity by simply adding BIG-IP devices to the device group and syncing the configuration, including the virtual IP address, to the new device. Note that all devices in the Sync-Only device group function as standalone devices and therefore do not share state, such as mirroring connections, persisting records, and so on.

You can configure all-active BIG-IP clustering in these ways:

Configuring the virtual server to use SNAT Auto Map

In the configuration example provided in this document, the pool members are on the internal network of the BIG-IP systems. However, because SNAT Auto Map selects the unique self IP address that is local to a device, the pool members can actually reside anywhere on the network, as long as they have a route back to those unique local self IP addresses.

Configuring the virtual server to use SNAT pools

- SNAT pools can provide segmentation of traffic per application, as well scale the amount of connections per pool member.
- In the configuration example provided in this document, the pool members are remote and located on the ECMP router's internal network. The virtual server leverages an iRule to select a unique SNAT pool for each device.
- Because the SNAT pool addresses in this configuration are shared across all devices in the BIG-IP device group, the SNAT addresses must be routable only (that is, not on a directly-connected network).
- All instances reference the same iRule, which selects the SNAT pool for each device. In this case, the server pool is on the ECMP router's internal network.

Important: *These configuration solutions make use of the administrative partitioning feature of the BIG-IP system. As you implement this configuration, it is essential that you create each BIG-IP object in the appropriate partition, as indicated within each object's configuration task.*

Chapter 2

Configuring Basic BIG-IP System Settings

- *Overview: Configuring basic system settings*

Overview: Configuring basic system settings

Whether you implement an ECMP-based all-active device group using SNAT Auto Map or by creating SNAT pools, you must first perform some basic Traffic Management Operating System® (TMOS) tasks. These basic tasks pertain to licensing and DNS confirmation, and NTP server configuration, followed by tasks to create VLANs and self IP addresses. Other tasks pertain to creating a BIG-IP® device group along with an administrative partition for local traffic objects.

After configuring these TMOS® objects, you can choose to implement either the SNAT Automap or the SNAT pool use case.

Task List

Confirming the contents of the BIG-IP license
Viewing the DNS server configuration
Specifying a list of NTP servers
Creating VLANs
Creating self IP addresses
Enabling dynamic routing protocols for route domain 0
Specifying an IP address for config sync
Establishing device trust
Creating a Sync-Only device group
Syncing the BIG-IP configuration to the device group
Creating an administrative partition
Changing the current partition

Confirming the contents of the BIG-IP license

On each BIG-IP® device that you intend to include in the cluster, you must verify that the license includes the advanced routing modules for dynamic routing.

Important: *You must perform this licensing task locally on each device that is to become a member of the device group.*

1. Access the BIG-IP system by logging in to the BIG-IP Configuration utility with your user credentials.
2. On the Main tab, click **System** > **License**.
3. In the Active Modules division of the properties, verify that Routing Bundle appears in the list of active modules.

Viewing the DNS server configuration

You perform this task to determine whether any DNS servers are specified on the BIG-IP® system for communication to other devices on the network.

Important: *You must perform this DNS task locally on each device that is to become a member of the device group.*

1. On the Main tab, click **System** > **Configuration** > **Device** > **DNS**.
   The DNS Device configuration screen opens.
2. View the **DNS Lookup Server List** settings to determine if any DNS servers have been configured on the BIG-IP system.

Specifying a list of NTP servers

If you use Dynamic Host Configuration Protocol (DHCP) to assign IP addresses to NTP servers, then before you perform this task, verify that you have configured a Domain Name System (DNS) server on the BIG-IP® system.

Network Time Protocol (NTP) synchronizes the clocks on a network of BIG-IP devices by means of a defined NTP server. This clock synchronization is required for successful operation of a BIG-IP device group. You can specify a list of the IP addresses of the defined NTP servers that you want the BIG-IP system to use when updating the time on BIG-IP systems on the network. Alternatively, you can specify a list of fully-qualified domain names.

Important: *You must perform this task locally on each BIG-IP device that is to be a member of the BIG-IP device group, and you must create the object in administrative partition* `Common`*.*

---

1. On the Main tab, click **System** > **Configuration** > **Device** > **NTP**.
   The NTP Device configuration screen opens.
2. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.
3. From the **Partition** list, confirm or select partition `Common`.
4. For the **Time Server List** setting, in the **Address** field, type the IP address of an NTP server that you want to add. Then click **Add**.

   Note: *If you are using Dynamic Host Configuration Protocol (DHCP) to assign IP addresses, then the BIG-IP system automatically populates the* ***Address*** *field with the fully-qualified domain name (FQDN) of the NTP server.*

   ---

5. Repeat the preceding step as needed.
6. Click **Update**.

Creating VLANs

VLANs represent a logical 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. For this implementation, F5 Networks recommends that you create three VLANs on each BIG-IP® device: a VLAN for the external network, a VLAN for the internal network, and a VLAN for high availability communications.

Important: *You must perform this task locally on each BIG-IP device that is to be a member of the BIG-IP device group, and you must create the object in administrative partition* `Common`*.*

---

1. On the Main tab, click **Network** > **VLANs**.
   The VLAN List screen opens.
2. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.

3. From the **Partition** list, confirm or select partition `Common`.
4. Click **Create**.
   The New VLAN screen opens.
5. In the **Name** field, type a unique name for the VLAN.
6. 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.
7. If you want to use Q-in-Q (double) tagging, use the **Customer Tag** setting to perform the following two steps. If you do not see the **Customer Tag** setting, your hardware platform does not support Q-in-Q tagging and you can skip this step.
   a) From the **Customer Tag** list, select **Specify**.
   b) Type a numeric tag, from 1-4094, for the VLAN.

   The customer tag specifies the inner tag of any frame passing through the VLAN.
8. For the **Interfaces** setting:
   a) From the **Interface** list, select an interface number.
   b) From the **Tagging** list, select **Tagged** or **Untagged**.

      Select **Tagged** when you want traffic for that interface to be tagged with a VLAN ID.
   c) If you specified a numeric value for the **Customer Tag** setting and from the **Tagging** list you selected **Tagged**, then from the **Tag Mode** list, select a value.
   d) Click **Add**.
   e) Repeat these steps for each interface that you want to assign to the VLAN.
9. If you want the system to verify that the return route to an initial packet is the same VLAN from which the packet originated, select the **Source Check** check box.
10. In the **MTU** field, retain the default number of bytes (**1500**).
11. From the **Configuration** list, select **Advanced**.
12. If you want to base redundant-system failover on VLAN-related events, select the **Fail-safe** check box.
13. From the **Auto Last Hop** list, select a value.
14. From the **CMP Hash** list, select a value.
15. To enable the **DAG Round Robin** setting, select the check box.
16. Click **Finished**.
    The screen refreshes, and displays the new VLAN in the list.

Creating self IP addresses

Self IP addresses enable the BIG-IP® system, and other devices on the network, to route application traffic through the associated VLAN. For this implementation, you perform this task on each BIG-IP device to create a unique static self IP address for each of the three VLANs (external, internal, and high availability). In this task, you replace any sample self IP names or IP addresses with the relevant self IP names or addresses for your network.

Note:

Important: *You must perform this task locally on each BIG-IP device that is to be a member of the BIG-IP device group, and you must create the self IP address in administrative partition* `Common`.

1. On the Main tab, click **Network** > **Self IPs**.
2. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.
3. From the **Partition** list, confirm or select partition `Common`.
4. Click **Create**.
   The New Self IP screen opens.
5. In the **Name** field, type a unique name for the self IP address.
   For example, for device `Bigip_1`, this name could be `ext_self_bigip1`, `int_self_bigip1`, or `ha_self_bigip1`.
6. In the **IP Address** field, type an IPv4 or IPv6 address.
   In our sample configuration for `Bigip_1`, this IP address is either `20.1.1.2`, `20.1.1.3`, or `20.1.1.5`.
7. In the **Netmask** field, type the full network mask for the specified IP address.

   For example, you can type `ffff:ffff:ffff:ffff:0000:0000:0000:0000` or `ffff:ffff:ffff:ffff::`.
8. From the **VLAN/Tunnel** list, select the VLAN to associate with this self IP address.
   - On the internal network, select the internal or high availability VLAN that is associated with an internal interface or trunk.
   - On the external network, select the external VLAN that is associated with an external interface or trunk.
9. If you are creating an external self IP address, use the **Port Lockdown** setting to add **TCP 179** to your current list of allowed ports for this self IP address.
   Port `179` represents the Border Gateway Protocol (BGP). Selecting port `179` gives BGP traffic coming from the ECMP router access to the BIG-IP device.
10. Click **Add**.
11. From the **Traffic Group** list, select **traffic-group-local-only (non-floating)**.
12. Click **Finished**.
    The screen refreshes, and displays the new self IP address.

The BIG-IP system can send and receive traffic through the specified VLAN.

Sample self IP addresses for BIG-IP devices

This table shows sample IP addresses for BIG-IP® devices, along with explanatory information.

BIG-IP device	Self IP address	Associated VLAN	Purpose
`Bigip_1`	`20.1.1.2`	External	The upstream ECMP router uses this address to load balance traffic to the virtual server on `Bigip_1`.
	`10.1.1.2`	Internal	This is the address that other device group members use when synchronizing a configuration to `Bigip_1`.
	`10.1.2.2`	High availability	This the address that other device group members use for high availability communications with `Bigip_1`.
`Bigip_2`	`20.1.1.3`	External	The upstream ECMP router uses this address to load balance traffic to the virtual server on `Bigip_2`.
	`10.1.1.3`	Internal	This is the address that other device group members use when synchronizing a configuration to `Bigip_2`.

BIG-IP device	Self IP address	Associated VLAN	Purpose
	10.1.2.3	High availability	This the address that other device group members use for high availability communications with Bigip_2.
Bigip_3	20.1.1.4	External	The upstream ECMP router uses this address to load balance traffic to the virtual server on Bigip_3.
	10.1.1.4	Internal	This is the address that other device group members use when synchronizing a configuration to Bigip_3.
	10.1.2.4	High availability	This the address that other device group members use for high availability communications with Bigip_3.

Enabling dynamic routing protocols for route domain 0

You perform this task to enable Border Gateway Protocol and any other dynamic routing protocols for route domain 0.

Important: *You must perform this task locally on each BIG-IP® device that is to be a member of the BIG-IP device group, and the current administrative partition must be set to* Common.

1. On the Main tab, click **Network** > **Route Domains**.
   The Route Domain List screen opens.
2. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.
3. From the **Partition** list, confirm or select partition Common.
4. In the Name column, click 0.
5. For the **Dynamic Routing Protocols** setting, from the **Available** list, select one or more protocol names and move them to the **Enabled** list.
   You can enable any number of listed protocols for this route domain.

   Important: *You must enable the BGP protocol.*
6. Click **Update**.
   The system displays the list of route domains on the BIG-IP system.

The dynamic routing protocols, including BGP, are enabled on the BIG-IP system.

Specifying an IP address for config sync

Before configuring the config sync address, verify that all devices in the device group are running the same version of BIG-IP® system software.

You perform this task to specify the IP address on the local device that other devices in the device group will use to synchronize their configuration objects to the local device.

Note: *You must perform this task locally on each device in the device group, and the current administrative partition must be set to* Common.

1. Confirm that you are logged in to the device you want to configure.

2. On the Main tab, click **Device Management** > **Devices**.
   This displays a list of device objects discovered by the local device.
3. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.
4. From the **Partition** list, confirm or select partition `Common`.
5. In the Name column, click the name of the device to which you are currently logged in.
6. From the **Device Connectivity** menu, choose ConfigSync.
7. For the **Local Address** setting, retain the displayed IP address or select another address from the list.
   F5 Networks recommends that you use the default value, which is the self IP address for the internal VLAN. This address must be a non-floating self IP address and not a management IP address.

   Important: *If the BIG-IP device you are configuring is accessed using Amazon Web Services, then the internal self IP address that you select must be an internal private IP address that you configured for this EC2 instance as the* ***Local Address****.*
8. Click **Update**.

After performing this task, the other devices in the device group can sync their configurations to the local device whenever a sync operation is initiated.

Establishing device trust

Before you begin this task, verify that:

- Each BIG-IP® device that is to be part of the local trust domain has a device certificate installed on it.
- The local device is designated as a certificate signing authority.

You perform this task to establish trust among devices on one or more network segments. Devices that trust each other constitute the *local trust domain*. A device must be a member of the local trust domain prior to joining a device group.

By default, the BIG-IP software includes a local trust domain with one member, which is the local device. You can choose any one of the BIG-IP devices slated for a device group and log into that device to add other devices to the local trust domain. For example, devices `Bigip_1`, `Bigip_2`, and `Bigip_3` each initially shows only itself as a member of the local trust domain. To configure the local trust domain to include all three devices, you can simply log into device `Bigip_1` and add devices `Bigip_2` and `Bigip_3` to the local trust domain; there is no need to repeat this process on devices `Bigip_2` and `Bigip_3`.

1. On the Main tab, click **Device Management** > **Device Trust**, and then either **Peer List** or **Subordinate List**.
2. Click **Add**.
3. Type a device IP address, administrator user name, and administrator password for the remote BIG-IP® device with which you want to establish trust. The IP address you specify depends on the type of BIG-IP device:
   - If the BIG-IP device is an appliance, type the management IP address for the device.
   - If the BIG-IP device is a VIPRION® device that is not licensed and provisioned for vCMP®, type the primary cluster management IP address for the cluster.
   - If the BIG-IP device is a VIPRION device that is licensed and provisioned for vCMP, type the cluster management IP address for the guest.
   - If the BIG-IP device is an Amazon Web Services EC2 device, type one of the Private IP addresses created for this EC2 instance.

4. Click **Retrieve Device Information**.
5. Verify that the certificate of the remote device is correct.
6. Verify that the management IP address and name of the remote device are correct.
7. Click **Finished**.

After you perform this task, the local device is now a member of the local trust domain. Also, the BIG-IP system automatically creates a special Sync-Only device group for the purpose of synchronizing trust information among the devices in the local trust domain, on an ongoing basis.

Repeat this task to specify each device that you want to add to the local trust domain.

Creating a Sync-Only device group

You perform this task to create a Sync-Only type of device group. When you create a Sync-Only device group, the BIG-IP® system can then automatically synchronize configuration data (such as security policies and acceleration applications) to the other devices in the group, even when some of those devices reside in another network.

Note: *You perform this task on any one BIG-IP device within the local trust domain; there is no need to repeat this process on the other devices in the device group.*

---

1. On the Main tab, click **Device Management** > **Device Groups**.
2. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.
3. From the **Partition** list, confirm or select partition `Common`.
4. On the Device Groups list screen, click **Create**.
   The New Device Group screen opens.
5. Type a name for the device group, select the device group type **Sync-Only**, and type a description for the device group.
6. From the **Configuration** list, select **Advanced**.
7. For the **Members** setting, select an IP address and host name from the **Available** list for each BIG-IP device that you want to include in the device group. Use the Move button to move the host name to the **Includes** list.

   The list shows any devices that are members of the device's local trust domain.
8. For the **Automatic Sync** setting, select or clear the check box:
   - Select the check box when you want the BIG-IP system to automatically sync the BIG-IP configuration data whenever a config sync operation is required. In this case, the BIG-IP system syncs the configuration data whenever the data changes on any device in the device group.
   - Clear the check box when you want to manually initiate each config sync operation. In this case, F5 networks recommends that you perform a config sync operation whenever configuration data changes on one of the devices in the device group.
9. For the **Full Sync** setting, select or clear the check box:
   - Select the check box when you want all sync operations to be full syncs. In this case, the BIG-IP system syncs the entire set of BIG-IP configuration data whenever a config sync operation is required.
   - Clear the check box when you want all sync operations to be incremental (the default setting). In this case, the BIG-IP system syncs only the changes that are more recent than those on the target device. When you select this option, the BIG-IP system compares the configuration data on each target device with the configuration data on the source device and then syncs the delta of each target-source pair.

If you enable incremental synchronization, the BIG-IP system might occasionally perform a full sync for internal reasons. This is a rare occurrence and no user intervention is required.

10. In the **Maximum Incremental Sync Size (KB)** field, retain the default value of `1024`, or type a different value.

    This value specifies the total size of configuration changes that can reside in the incremental sync cache. If the total size of the configuration changes in the cache exceeds the specified value, the BIG-IP system performs a full sync whenever the next config sync operation occurs.

11. Click **Finished**.

You now have a Sync-Only type of device group containing BIG-IP devices as members.

Syncing the BIG-IP configuration to the device group

Before you sync the configuration, verify that the devices targeted for config sync are members of a device group and that device trust is established.

This task synchronizes the BIG-IP® configuration data from the local device to the devices in the device group. This synchronization ensures that devices in the device group operate properly.

Note: *You perform this task on only one device in the device group.*

---

1. On the Main tab, click **Device Management** > **Overview**.
2. In the Device Groups area of the screen, in the Name column, select the name of the relevant device group.
   The screen expands to show a summary and details of the sync status of the selected device group, as well as a list of the individual devices within the device group.
3. In the Devices area of the screen, in the Sync Status column, select the device that shows a sync status of `Changes Pending`.
4. In the Sync Options area of the screen, select **Sync Device to Group**.
5. Click **Sync**.
   The BIG-IP system syncs the configuration data of the selected device in the Device area of the screen to the other members of the device group.

The BIG-IP configuration data is replicated on each device in the device group.

Creating an administrative partition

You perform this task to create an administrative partition that is associated with a BIG-IP® Sync-Only device group. An *administrative partition* creates an access control boundary for users and applications. When you create a partition for this implementation, you assign the Sync-Only device group as an attribute of the folder that corresponds to the partition. For example, you can create a partition named `Spanned_VIP` and set its **Device Group** setting to `my_sync_only_dg`. In this case, the system automatically creates a folder named `/Spanned_VIP`, and any local traffic objects that you create in that folder will be synchronized to the devices in device group `my_sync_only_dg`, whenever a config sync operation occurs.

Note: *You perform this task on only one device in the device group; there is no need to repeat this process on the other device group members.*

---

1. On the Main tab, expand **System** and click **Users**.
   The Users List screen opens.

2. On the menu bar, click **Partition List**.
3. Click **Create**.
   The New Partition screen opens.
4. In the **Partition Name** field, type a unique name for the partition.
   An example of a partition name is `Spanned_VIP`.
5. Type a description of the partition in the **Description** field.
   This field is optional.
6. For the **Device Group** setting, clear the **Inherit device group from root folder** check box and from the list, select the name of the Sync-Only device group.
7. From the **Traffic Group** list, choose **None**.
8. Click **Finished**.

After creating the partition, you can create local traffic objects within the partition that the BIG-IP® system will synchronize to the other devices in the BIG-IP device group.

Changing the current partition

You perform this task to change the current administrative partition on the BIG-IP® system. You change the partition when you want BIG-IP configuration objects that you create to reside in the folder that corresponds to the partition. For example, if the current partition is set to `Common`, but you want to create a load balancing pool and virtual server in folder `/Spanned_VIP` instead of `/Common`, you can switch the current partition to partition `Spanned_VIP`. Any configuration objects you subsequently create will reside in folder `/Spanned_VIP` and will be synchronized to the Sync-Only device group defined as an attribute of that folder.

1. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.
2. From the **Partition** list, select the partition in which you want to create local traffic objects.

After you perform this task, any configuration objects that you create reside in the folder corresponding to the selected partition. For example, if you selected partition `Spanned_VIP`, subsequent objects such as a load balancing pool and a virtual server will reside in folder `/Spanned_VIP`.

Chapter 3

Use Case 1: Creating a Configuration that Uses SNAT Auto Map

- *Using SNAT Auto Map*
- *Implementation result*

Using SNAT Auto Map

One of the ways that you can set up all-active clustering of BIG-IP® devices is through the use of SNAT Auto Map. This example includes an ECMP-enabled router on the BIG-IP external network and a load balancing pool on the internal network. Each device in the device group has the same virtual IP address and provides a unique, static self IP address for the next-hop route to the virtual server. Furthermore, by enabling SNAT Auto Map on the virtual server, each server response returns through the originating device by way of the unique self IP address on its way back to the client.

This illustration shows an example of this configuration.

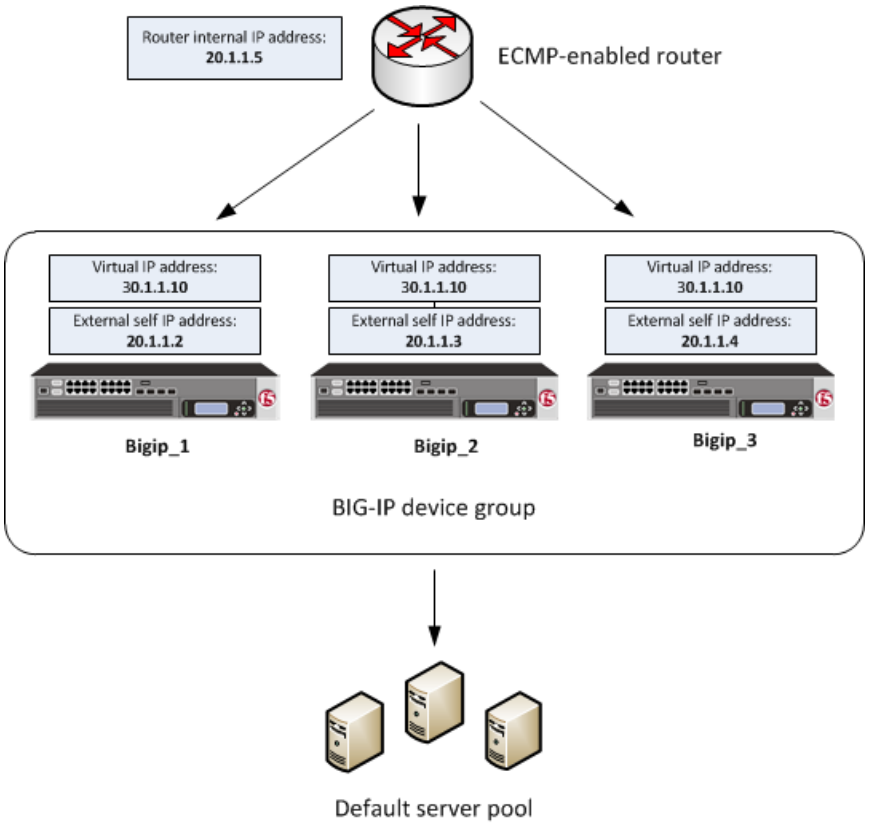

Figure 1: BIG-IP system clustering using ECMP with SNAT Auto Map

Creating a load balancing pool
Creating a virtual server
Confirming virtual address exclusion from a traffic group
Syncing the BIG-IP configuration to the device group
Configuring the BGP protocol

Creating a load balancing pool

You can create a load balancing pool to efficiently distribute the load on your server resources. A *load balancing pool* is a logical representation of the set of servers grouped together on the network to process traffic. After you synchronize the configuration later to the other BIG-IP® devices in the device group, the same load balancing pool is configured on all device group members.

Note: *You perform this task on only one device in the device group. You will later synchronize this configuration to the other devices in the device group.*

1. On the Main tab, click **Local Traffic** > **Pools**.
   The Pool List screen opens.
2. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.
3. From the **Partition** list, select the partition in which you want to create local traffic objects.
4. Click **Create**.
   The New Pool screen opens.
5. In the **Name** field, type a unique name for the pool.
6. For the **Health Monitors** setting, in the **Available** list, select a monitor type, and click **<<** to move the monitor to the **Active** list.

   Tip: *Hold the Shift or Ctrl key to select more than one monitor at a time.*

7. From the **Load Balancing Method** list, select how the system distributes traffic to members of this pool.
   The default is **Round Robin**.
8. For the **Priority Group Activation** setting, specify how to handle priority groups:
   - Select **Disabled** to disable priority groups. This is the default option.
   - Select **Less than**, and in the **Available Members** field type the minimum number of members that must remain available in each priority group in order for traffic to remain confined to that group.
9. Using the **New Members** setting, add each resource that you want to include in the pool:
   a) In the **Node Name** field, type a name for the node portion of the pool member.
      This step is optional.
   b) In the **Address** field, type an IP address.
      This address will reside on the internal subnet of the BIG-IP devices.
   c) In the **Service Port** field, type a port number, or select a service name from the list.
   d) In the **Priority** field, type a priority number.
      This step is optional.
   e) Click **Add**.
10. Click **Finished**.

The load balancing pool appears in the Pools list.

Creating a virtual server

You perform this task to provide a destination for application traffic coming into the BIG-IP® system from an ECMP-enabled router on the network. After you synchronize the configuration later to the other devices in the device group, the same virtual server is configured on all of the BIG-IP devices in the device group.

Note: *You perform this task on only one device in the device group. You will later synchronize this configuration to the other devices in the device group.*

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.

The Virtual Server List screen opens.

2. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.
3. From the **Partition** list, select the partition in which you want to create local traffic objects.
4. Click the **Create** button.
   The New Virtual Server screen opens.
5. In the **Name** field, type a unique name for the virtual server.
6. From the **Type** list, select **Standard**.
7. In the **Destination Address** field, type the IP address in CIDR format.

   The supported format is address/prefix, where the prefix length is in bits. For example, an IPv4 address/prefix is `10.0.0.1` or `10.0.0.0/24`, and an IPv6 address/prefix is `ffe1::0020/64` or `2001:ed8:77b5:2:10:10:100:42/64`. When you use an IPv4 address without specifying a prefix, the BIG-IP® system automatically uses a `/32` prefix.

   Note: *This address must be on a separate network available only through routing (instead of through a directly-connected network).*

   In our example, this address is `30.1.1.10`.
8. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
9. From the **Source Address Translation** list, select **Auto Map**.
10. In the Resources area of the screen, from the **Default Pool** list, select the relevant pool name.
11. Configure any other settings as needed.
12. Click **Finished**.

The virtual server appears in the list of existing virtual servers on the Virtual Server List screen.

Confirming virtual address exclusion from a traffic group

You perform this task to confirm that the virtual address is excluded from being a member of any traffic group on a BIG-IP® device in the device group. A virtual address inherits its traffic group membership from the partition in which the virtual address resides.

Important: *You perform this task on only one device in the device group. You will later synchronize this configuration to the other devices in the device group.*

---

1. On the Main tab, click **Local Traffic** > **Virtual Servers** > **Virtual Address List**.
   The Virtual Address List screen opens.
2. In the Name column, click the virtual IP address that the BIG-IP system created when you created the virtual server.
   This displays the properties of that virtual address.
3. For the **Traffic Group** setting, confirm that the value is set to **None**.
4. Click **Update**.

After you perform this task, the virtual IP address is no longer a member of any traffic group on the BIG-IP device.

Syncing the BIG-IP configuration to the device group

Before you sync the configuration, verify that the devices targeted for config sync are members of a device group and that device trust is established.

This task synchronizes the BIG-IP® configuration data from the local device to the devices in the device group. This synchronization ensures that devices in the device group operate properly.

Note: *You perform this task on only one device in the device group.*

1. On the Main tab, click **Device Management** > **Overview**.
2. In the Device Groups area of the screen, in the Name column, select the name of the relevant device group.
   The screen expands to show a summary and details of the sync status of the selected device group, as well as a list of the individual devices within the device group.
3. In the Devices area of the screen, in the Sync Status column, select the device that shows a sync status of `Changes Pending`.
4. In the Sync Options area of the screen, select **Sync Device to Group**.
5. Click **Sync**.
   The BIG-IP system syncs the configuration data of the selected device in the Device area of the screen to the other members of the device group.

The BIG-IP configuration data is replicated on each device in the device group.

Configuring the BGP protocol

Before performing this task, verify that you have permission to access the Bash shell.

Perform this task when you want to configure the Border Gateway Protocol (BGP) dynamic routing protocol.

Important: *You must perform this task locally on each BIG-IP® device.*

1. Open a console window, or an SSH session using the management port, on a BIG-IP device.
2. Log in to the BIG-IP system using your user credentials.
3. At the Bash command prompt, type `imish`.
   This command invokes the IMI shell.
4. Type `enable`.
5. Type `configure terminal`.
6. Type the relevant configuration commands.

   Note: *See the relevant sample BGP configuration in this document.*

7. Type `copy running-config startup-config`.
   The startup configuration file is `/config/zebos/rd0/ZebOS.conf`.
8. At the command prompt, type `disable`.
9. Type `exit`.

Sample BGP configuration using SNAT Auto Map

This example shows part of an example of Border Gateway Protocol (BGP) configuration on a BIG-IP® device that accepts traffic from an upstream ECMP-enabled router.

```
router bgp 65001
bgp router-id 20.1.1.2
redistribute kernel route-map f5-to-upstream
neighbor 20.1.1.5 remote-as 65000
!
ip prefix-list RHI-routes seq 5 permit 30.1.1.10/32
!
route-map f5-to-upstream permit 10
match ip address prefix-list RHI-routes
set ip next-hop to 20.1.1.2 primary
!
line con 0
login
line vty 0 39
login
!
end
```

Configuration entry	Description
`bgp router-id 20.1.1.2`	The `bgp router-id` value is the self IP address for the external VLAN on device `Bigip_1`. This address must be unique within the BGP configuration on each BIG-IP device in the device group.
`neighbor 20.1.1.5`	The `neighbor` value is the IP address of the ECMP-enabled router. You must repeat the `neighbor` statement for each upstream router associated with a BIG-IP device. These `neighbor` statements are the same within the BGP configuration on each BIG-IP device in the device group.
`ip prefix-list`	The `ip prefix-list` entry specifies that the virtual IP address `30.1.1.10/32` is allowed to be advertised.
`set ip next-hop 20.1.1.2`	The `set ip next-hop` value is the self IP address for the external VLAN on device `Bigip_1`. This next-hop address is used for traffic that is destined for the virtual IP address and potentially the specified SNAT pool address. This address must be unique within the BGP configuration on each BIG-IP device in the device group.

Implementation result

After following the instructions in this implementation, you now have a three-member BIG-IP® device group, where the same virtual server resides on each device, and each device is configured for dynamic routing using the Border Gateway Protocol (BGP). Also, the external self IP address that you created on each device is configured to allow the upstream ECMP-enabled router to send traffic through port 179 on each BIG-IP device.

With this configuration, when application traffic comes through the ECMP-enabled router, the router can use an algorithm to choose the best equal-cost path to any one of the BIG-IP devices in the device group. If any BIG-IP device becomes unavailable, the ECMP algorithm causes the ECMP-enabled router to forward that traffic to another device in the device group. Furthermore, each BIG-IP device has an administrative

partition whose local traffic objects are synchronized to the devices in the Sync-Only device group. All devices in the device group use the default load balancing pool to process application traffic.

Chapter 4

Use Case 2: Creating a Configuration that Uses a SNAT Pool

- *Using a SNAT pool*
- *Implementation result*

Using a SNAT pool

One of the ways that you can set up all-active clustering of BIG-IP® devices, using an ECMP-enabled router, is through the use of SNAT pools. You can use SNAT pools to provide segmentation of traffic per application, as well scale the amount of connections per pool member. To use SNAT pools, you first create a unique SNAT pool for each device in the BIG-IP device group and then create an iRule that selects a SNAT pool per device.

With this SNAT pool configuration, the server pool members return traffic to the SNAT address or addresses of the originating BIG-IP cluster device instead of to the unique self IP address (as is the case with the SNAT Auto Map configuration).

This illustration shows an example of this configuration.

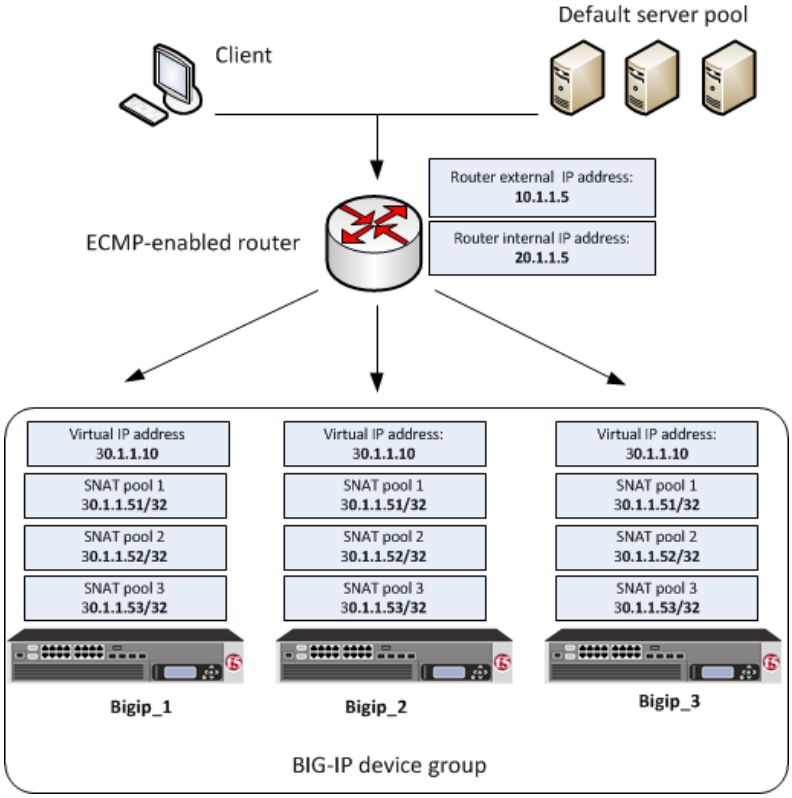

Figure 2: BIG-IP system clustering using ECMP with a SNAT pool

Creating a load balancing pool
Defining a route to the server
Creating SNAT pools
Creating a string data group
Creating an iRule for SNAT pool selection
Creating a virtual server
Confirming virtual address exclusion from a traffic group
Syncing the BIG-IP configuration to the device group
Configuring the BGP protocol

Creating a load balancing pool

You can create a load balancing pool to efficiently distribute the load on your server resources. A *load balancing pool* is a logical representation of the set of servers grouped together on the network to process traffic. After you synchronize the configuration later to the other BIG-IP® devices in the device group, the same load balancing pool is configured on all device group members.

Note: *You perform this task on only one device in the device group. You will later synchronize this configuration to the other devices in the device group.*

---

1. On the Main tab, click **Local Traffic** > **Pools**.
   The Pool List screen opens.
2. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.
3. From the **Partition** list, select the partition in which you want to create local traffic objects.
4. Click **Create**.
   The New Pool screen opens.
5. In the **Name** field, type a unique name for the pool.
   An example of a pool name is `external-pool`.
6. For the **Health Monitors** setting, in the **Available** list, select a monitor type, and click **<<** to move the monitor to the **Active** list.

   Tip: *Hold the Shift or Ctrl key to select more than one monitor at a time.*

   ---

7. From the **Load Balancing Method** list, select how the system distributes traffic to members of this pool.

   The default is **Round Robin**.
8. For the **Priority Group Activation** setting, specify how to handle priority groups:
   - Select **Disabled** to disable priority groups. This is the default option.
   - Select **Less than**, and in the **Available Members** field type the minimum number of members that must remain available in each priority group in order for traffic to remain confined to that group.
9. Using the **New Members** setting, add the resource that you want to include in the pool:
   a) In the **Node Name** field, type a name for the node portion of the pool member.
      This step is optional.
   b) In the **Address** field, type an IP address.
      This address will reside on the external network of the ECMP-enabled upstream router.
   c) In the **Service Port** field, type a port number, or select a service name from the list.
   d) In the **Priority** field, type a priority number.
      This step is optional.
   e) Click **Add**.
10. Click **Finished**.

The load balancing pool appears in the Pools list.

Defining a route to the server

You must define a route on the local BIG-IP® system for sending traffic to the server. When you perform this task, the destination IP address is the address of a pool member. The gateway IP address is the external IP address of the ECMP-enabled upstream router.

Note: *You perform this task on only one device in the device group. You will later synchronize this configuration to the other devices in the device group.*

1. On the Main tab, click **Network** > **Routes**.
2. Click **Add**.
   The New Route screen opens.
3. In the **Destination** field, type the network of the destination server.
   In our example, this address is `10.1.1.0`.
4. In the **Netmask** field, type the network mask for the destination IP address.
5. From the **Resource** list, select **Use Gateway**.
   The gateway represents a next-hop or last-hop address in the route.
6. From the **Gateway Address** list, select **IP Address**, and then type the external address of the ECMP-enabled upstream router, `20.1.1.4`.

Creating SNAT pools

You perform this task to create three separate SNAT pools on the BIG-IP® system. A SNAT pool consists of any IP addresses that you want the BIG-IP system to use as a SNAT translation address. For this implementation, each SNAT pool will contain only one address, and this address is unique.

Note: *You perform this task on only one device in the device group. You will later synchronize this configuration to the other devices in the device group.*

1. On the Main tab, click **Local Traffic** > **Address Translation** > **SNAT Pool List**.
   The SNAT Pool List screen displays a list of existing SNATs.
2. Click **Create**.
3. In the **Name** field, type a name for the SNAT pool.
   An example of a name is `snat-pool-1`.
4. For the **Member List** setting:
   a) In the **IP Address** field, type an IP address.
      The BIG-IP system will use this address as a SNAT translation address.

      Important: *This address must NOT be on a directly-connected network.*

   b) Click **Add**.
5. Use the **Repeat** button to create two other SNAT pools, each with a unique SNAT translation address, and then click **Finished**.

After performing this task, three SNAT pools reside on the BIG-IP system. Each SNAT pool contains a different SNAT translation address.

Creating a string data group

You can create a data group that maps each BIG-IP ®device in a device group to a separate SNAT pool containing a unique SNAT translation address.

Note: *You perform this task on only one device in the device group. You will later synchronize this configuration to the other devices in the device group.*

1. On the Main tab, click **Local Traffic** > **iRules** > **Data Group List**.
   The Data Group List screen opens, displaying a list of data groups on the system.
2. Click **Create**.
   The New Data Group screen opens.
3. In the **Name** field, type a unique name for the data group.
   An example of a data group name is `cluster_snatpool_dg`.
4. From the **Type** list, select **String**.
5. Using the **String Records** setting, create entries consisting of a BIG-IP device name and a SNAT pool name:
   a) In the **String** field, type the fully-qualified domain name of a BIG-IP system in the device group (using lowercase characters only).
   An example of an entry is `bigip_1.ecmp.test.com`.
   b) In the **Value** field, type the name of a SNAT pool.
   c) Click **Add**.
   d) Repeat these steps for each BIG-IP device and SNAT pool that you want to include in this data group.

   The result should look similar to this:

```
bigip_1.ecmp.test.com:= snat-pool-1
bigip_2.ecmp.test.com:= snat-pool-2
bigip_2.ecmp.test.com:= snat-pool-2
```

6. Click **Finished**.
   The new data group appears in the list of data groups.

After you perform this task, the BIG-IP system contains a data group that associates each BIG-IP device in the device group with a unique SNAT pool.

Creating an iRule for SNAT pool selection

You perform this task to create an iRule® that selects the correct SNAT pool for each BIG-IP® device. This provides a way for the ECMP-enabled router to select a custom IP address instead of a BIG-IP local self IP address.

Note: *You perform this task on only one device in the device group. You will later synchronize this configuration to the other devices in the device group.*

1. On the Main tab, click **Local Traffic** > **iRules**.
2. Click **Create**.
3. In the **Name** field, type a 1- to 31-character name, such as `snat-pool-select`.

4. In the **Definition** field, type the syntax for the iRule using Tool Command Language (Tcl) syntax.
   For complete and detailed information about iRules syntax, see the F5 Networks DevCentral web site `http://devcentral.f5.com`.
5. Click **Finished**.

Example of an iRule for SNAT pool selection

This example shows an iRule that selects the correct SNAT pool on a BIG-IP® device in a device group.

```
when RULE_INIT {
       # Log debug messages to /var/log/ltm? 1=yes, 0=no
       set static::debug_rule 0
       #v11.1.0 HF3 - v11.3.x
                    #set static::local_machine_name $static::tcl_platform(machine)
       #v11.4.0 - Current
                    set static::local_machine_name $::tcl_platform(machine)
}
when CLIENT_ACCEPTED {
   if { $static::debug_rule } { log local0.info "local_machine_name is
$static::local_machine_name" }
   set cluster_snatpool [ class match -value -- $static::local_machine_name equals
cluster_snatpool_dg ]
 # Check to see if there's a match in the datagroup
   if { $cluster_snatpool ne "" } {
         if { $static::debug_rule } { log local0.info "Attempting to use snatpool
$cluster_snatpool" }
              # Try to assign snatpool. Make sure snatpool itself exists
              if { [catch {snatpool $cluster_snatpool } result] }{
              # Log a message with the snatpool name which failed.
              log local0.err "Error: Client: [IP::client_addr]:[TCP::client_port]: Error
assigning snatpool \"$cluster_snatpool\": \$result: $result"
                    }
    }
}
```

Creating a virtual server

You perform this task to provide a destination for application traffic coming into the BIG-IP® system from an ECMP-enabled router on the network. After you synchronize the configuration later to the other devices in the device group, the same virtual server is configured on all of the BIG-IP devices in the device group.

Note: *You perform this task on only one device in the device group. You will later synchronize this configuration to the other devices in the device group.*

---

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen opens.
2. Locate the **Partition** list in the upper right corner of the BIG-IP Configuration utility screen, to the left of the **Log out** button.
3. From the **Partition** list, select the partition in which you want to create local traffic objects.
4. Click the **Create** button.
   The New Virtual Server screen opens.
5. In the **Name** field, type a unique name for the virtual server.
6. From the **Type** list, select **Standard**.
7. In the **Destination Address** field, type the IP address in CIDR format.

The supported format is address/prefix, where the prefix length is in bits. For example, an IPv4 address/prefix is `10.0.0.1` or `10.0.0.0/24`, and an IPv6 address/prefix is `ffe1::0020/64` or `2001:ed8:77b5:2:10:10:100:42/64`. When you use an IPv4 address without specifying a prefix, the BIG-IP® system automatically uses a `/32` prefix.

Note: *This address must be on a separate network available only through routing (instead of through a directly-connected network).*

In our example, this address is `30.1.1.10`.

8. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
9. From the **Source Address Translation** list, select **None**.
10. In the Resources area of the screen, from the **Default Pool** list, select the name of the pool you created previously.
    In our example, the name of this pool is `external-pool`.
11. For the **Related iRules** setting, from the **Available** list, select the name of the iRule that you want to assign, and move the name to the **Enabled** list.
    In our example, the name of this iRule is `snat-pool-select`.
12. Configure any other settings as needed.
13. Click **Finished**.

The virtual server appears in the list of existing virtual servers on the Virtual Server List screen.

Confirming virtual address exclusion from a traffic group

You perform this task to confirm that the virtual address is excluded from being a member of any traffic group on a BIG-IP® device in the device group. A virtual address inherits its traffic group membership from the partition in which the virtual address resides.

Important: *You perform this task on only one device in the device group. You will later synchronize this configuration to the other devices in the device group.*

1. On the Main tab, click **Local Traffic** > **Virtual Servers** > **Virtual Address List**.
   The Virtual Address List screen opens.
2. In the Name column, click the virtual IP address that the BIG-IP system created when you created the virtual server.
   This displays the properties of that virtual address.
3. For the **Traffic Group** setting, confirm that the value is set to **None**.
4. Click **Update**.

After you perform this task, the virtual IP address is no longer a member of any traffic group on the BIG-IP device.

Syncing the BIG-IP configuration to the device group

Before you sync the configuration, verify that the devices targeted for config sync are members of a device group and that device trust is established.

This task synchronizes the BIG-IP® configuration data from the local device to the devices in the device group. This synchronization ensures that devices in the device group operate properly.

Note: *You perform this task on only one device in the device group.*

1. On the Main tab, click **Device Management** > **Overview**.
2. In the Device Groups area of the screen, in the Name column, select the name of the relevant device group.
   The screen expands to show a summary and details of the sync status of the selected device group, as well as a list of the individual devices within the device group.
3. In the Devices area of the screen, in the Sync Status column, select the device that shows a sync status of `Changes Pending`.
4. In the Sync Options area of the screen, select **Sync Device to Group**.
5. Click **Sync**.
   The BIG-IP system syncs the configuration data of the selected device in the Device area of the screen to the other members of the device group.

The BIG-IP configuration data is replicated on each device in the device group.

Configuring the BGP protocol

Before performing this task, verify that you have permission to access the Bash shell.

Perform this task when you want to configure the Border Gateway Protocol (BGP) dynamic routing protocol.

Important: *You must perform this task locally on each BIG-IP® device.*

1. Open a console window, or an SSH session using the management port, on a BIG-IP device.
2. Log in to the BIG-IP system using your user credentials.
3. At the Bash command prompt, type `imish`.
   This command invokes the IMI shell.
4. Type `enable`.
5. Type `configure terminal`.
6. Type the relevant configuration commands.

   Note: *See the relevant sample BGP configuration in this document.*

7. Type `copy running-config startup-config`.
   The startup configuration file is `/config/zebos/rd0/ZebOS.conf`.
8. At the command prompt, type `disable`.
9. Type `exit`.

Sample BGP configuration using a SNAT pool

This example shows part of a Border Gateway Protocol (BGP) configuration on a BIG-IP® device that accepts traffic from an upstream ECMP-enabled router. In this example, a static route or static routes are being used to distribute the unique SNAT pool addresses associated with each BIG-IP device.

```
router bgp 65001
bgp router-id 20.1.1.2
redistribute kernel route-map f5-to-upstream
redistribute static route-map f5-to-upstream
neighbor 20.1.1.5 remote-as 65000
```

```
!
ip route 30.1.1.51/32 tmm0
!
ip prefix-list RHI-routes seq 5 permit 30.1.1.10/32
ip prefix-list RHI-routes seq 10 permit 30.1.1.51/32
!
route-map f5-to-upstream permit 10
match ip address prefix-list RHI-routes
set ip next-hop 20.1.1.2 primary
!
line con 0
 login
line vty 0 39
 login
!
end
```

Configuration entry	Description
`bgp router-id 20.1.1.2`	The `bgp router-id` value is the self IP address for the external VLAN on device `Bigip_1`. This address must be unique within the BGP configuration on each BIG-IP device in the device group.
`redistribute static route-map f5-to-upstream`	This entry ensures that the system advertises the SNAT pool address specified in the `ip route` entry.
`neighbor 20.1.1.5`	The `neighbor` value is the IP address of the ECMP-enabled router. You must repeat the `neighbor` statement for each upstream router associated with a BIG-IP device. These `neighbor` statements are the same within the BGP configuration on each BIG-IP device in the device group.
`ip route 30.1.1.51/32`	The `ip route` value is the translation address contained in SNAT pool `snat-pool-1`. Setting `ip route` to the SNAT pool address ensures that the system advertises this address. If the SNAT pool in your own configuration contains more than one translation address, you must include an `ip route` entry for each translation address in the SNAT pool. This address must be unique within the BGP configuration for each device in the device group.
`ip prefix-list`	The `ip prefix-list` entry specifies that the virtual IP address `30.1.1.10/32` and the SNAT address `30.1.1.51/32` are allowed to be advertised.
`set ip next-hop 20.1.1.2`	The `set ip next-hop` value is the self IP address for the external VLAN on device `Bigip_1`. This next-hop address is used for traffic that is destined for the virtual IP address and potentially the specified SNAT pool address. This address must be unique within the BGP configuration on each BIG-IP device in the device group.

Implementation result

After following the instructions in this implementation, you now have a three-member BIG-IP® device group, where the same virtual server resides on each device, and each device is configured for dynamic routing using the Border Gateway Protocol (BGP). Also, the SNAT pool that you created on each device allows the upstream ECMP-enabled router to send traffic through port 179 on each BIG-IP device.

With this configuration, when application traffic comes through the ECMP-enabled router, the router can use an algorithm to select the best equal-cost path to any one of the BIG-IP devices in the device group. If any BIG-IP device becomes unavailable, the ECMP algorithm causes the ECMP-enabled router to forward that traffic to another device in the device group. Furthermore, each BIG-IP device has an administrative partition whose local traffic objects are synchronized to the devices in the Sync-Only device group. All devices in the device group use the default load balancing pool, which contains a single server on the ECMP router's internal network, to process application traffic.

Index

A

administrative partitions
  changing *30*
  creating *29*
all-active clusters
  using ECMP protocol *20*, *22*
application traffic
  isolating on network *29*
automatic synchronization
  enabling and disabling *28*

B

BGP protocol
  configuring *35*, *46*
  sample configuration file *36*, *46*

C

clustering
  and SNAT Auto Map *32*
config sync addresses
  specifying *26*
configuration synchronization
  syncing to group *29*, *35*, *45*
connections
  scaling per pool member *40*

D

data groups
  creating *43*
device discovery
  for device trust *27*
device groups
  creating *28*
devices
  clustering *20*, *22*
device trust
  establishing *27*
DNS resolution
  specifying DNS server *22*
dynamic routing protocols
  configuring *35*, *46*

E

ECMP-enabled cluster configuration
  results of *36*, *47*
ECMP protocol
  with device clustering *20*, *22*
equal-cost paths
  to BIG-IP system addresses *20*, *22*

F

folders
  changing *30*

H

health monitors
  assigning to pools *32*, *41*

I

IMI shell *35*, *46*
iRules
  example of *44*
  for selecting SNAT pools *43*

L

license confirmation *22*
local trust domain
  and device groups *28*
  defined *27*

M

monitors
  assigning to pools *32*, *41*

N

NTP servers
  defining *23*

P

partitions
See also administrative partitions
  changing *30*
  See also administrative partitions
performance monitors
  assigning to pools *32*, *41*
pools
  creating *32*, *41*

R

route domains
  creating *26*
routes
  defining *42*

S

self IP addresses
  and VLANs *24*

self IP addresses *(continued)*
creating *24*
examples of *25*
SNAT
creating *42*
SNAT Auto Map
purpose of *32*
SNAT pools
use of *40*
using iRule to select *44*
string data groups
creating *43*
Sync-Only device groups
creating *28*

T

trust domains
and local trust domain *27*

V

virtual addresses
removing from traffic group *34*, *45*
virtual servers
creating ECMP-related traffic *44*
creating for HTTP traffic *33*
VLANs
and self IP addresses *24*
creating *23*

X

x509 certificates
for device trust *27*