

BIG-IP[®] System: iRules[®] Concepts

Version 11.6



Table of Contents

Legal Notices	5
Acknowledgments	7
Chapter 1: Introduction to iRules	19
What is an iRule?.....	20
iRule commands.....	20
Event declarations.....	21
Operators.....	21
Creating an iRule.....	21
Chapter 2: iRule Commands	23
iRule command types.....	24
The pool command.....	24
The node command.....	25
Commands that select a pool of cache servers.....	25
The HTTP::redirect command.....	26
The snat and snatpool commands.....	26
Chapter 3: iRule Evaluation	27
About iRule evaluation.....	28
Event types.....	28
iRule context.....	28
iRules assignment to a virtual server.....	29
Chapter 4: iRules and Administrative Partitions	31
iRules and administrative partitions.....	32
Chapter 5: iRules and Local Traffic Profiles	33
iRules and profiles.....	34
The profile command.....	34
Commands that override profile settings.....	34
Chapter 6: Data Groups	35
About data groups.....	36
About the class match command.....	36
Storage options.....	36
About file import for data groups.....	37

Chapter 7: iFiles.....	39
About file import for iRules.....	40
Importing a file for an iRule.....	40
Viewing a list of imported files.....	40
About iFiles.....	40
Viewing a list of iFiles.....	41
Creating an iFile.....	41
iRule commands for referencing an iFile.....	41

Legal Notices

Publication Date

This document was published on June 1, 2015.

Publication Number

MAN-0543-00

Copyright

Copyright © 2014-2015, F5 Networks, Inc. All rights reserved.

F5 Networks, Inc. (F5) believes the information it furnishes to be accurate and reliable. However, F5 assumes no responsibility for the use of this information, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright, or other intellectual property right of F5 except as specifically described by applicable user licenses. F5 reserves the right to change specifications at any time without notice.

Trademarks

AAM, Access Policy Manager, Advanced Client Authentication, Advanced Firewall Manager, Advanced Routing, AFM, Application Acceleration Manager, Application Security Manager, APM, 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 Certified [DESIGN], F5 Networks, F5 SalesXchange [DESIGN], F5 Synthesis, f5 Synthesis, F5 Synthesis [DESIGN], F5 TechXchange [DESIGN], Fast Application Proxy, Fast Cache, FirePass, Global Traffic Manager, GTM, GUARDIAN, iApps, IBR, iCall, Intelligent Browser Referencing, Intelligent Compression, IPv6 Gateway, iControl, iHealth, iQuery, iRules, iRules OnDemand, iSession, L7 Rate Shaping, LC, Link Controller, LineRate, LineRate Systems [DESIGN], Local Traffic Manager, LROS, LTM, Message Security Manager, MobileSafe, MSM, OneConnect, Packet Velocity, PEM, Policy Enforcement Manager, Protocol Security Manager, PSM, Real Traffic Policy Builder, SalesXchange, ScaleN, SDAC (except in Japan), SDC, Signalling Delivery Controller, Solutions for an application world, Software Designed Applications Services, SSL Acceleration, StrongBox, SuperVIP, SYN Check, TCP Express, TDR, TechXchange, TMOS, TotALL, Traffic Management Operating System, Traffix (except Germany), Traffix [DESIGN] (except Germany), Transparent Data Reduction, UNITY, VAULT, vCMP, VE F5 [DESIGN], Versafe, Versafe [DESIGN], VIPRION, Virtual Clustered Multiprocessing, WebSafe, 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.

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.

Acknowledgments

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 software developed by www.samba.org, which is protected under the GNU Public License, copyright ©2007.

This product includes software from Allan Jardine, distributed under the MIT License.

This product includes software from Trent Richardson, distributed under the MIT License.

This product includes vmbus drivers distributed by Microsoft Corporation.

This product includes software from Cavium.

This product includes software from Webroot, Inc.

This product includes software from Maxmind, Inc.

This product includes software from OpenVision Technologies, Inc. Copyright ©1993-1996, OpenVision Technologies, Inc. All Rights Reserved.

This product includes software developed by Matt Johnson, distributed under the MIT License. Copyright ©2012.

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 from NLnetLabs. Copyright ©2001-2006. All rights reserved.

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

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- 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.
- Neither the name of NLnetLabs nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS 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 COPYRIGHT OWNER 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 GRand Unified Bootloader (GRUB) software developed under the GNU Public License, copyright ©2007.

Acknowledgments

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 gdtf.c copyright ©1999, 2000, 2001, 2002 John Ellson (ellson@lucent.com).
- Portions relating to gdf.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.oracle.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 the 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.

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 licensed from Rémi Denis-Courmont under the GNU Library General Public License. Copyright ©2006 - 2011.

This product includes software developed by jQuery Foundation and other contributors, distributed under the MIT License. Copyright ©2014 jQuery Foundation and other contributors (<http://jquery.com/>).

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 Trent Richardson, distributed under the MIT License. Copyright ©2012 jQuery Foundation and other contributors (<http://jquery.com/>).

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 Allan Jardine, distributed under the MIT License. Copyright ©2008 - 2012, Allan Jardine, all rights reserved, jQuery Foundation and other contributors (<http://jquery.com/>).

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 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.

Acknowledgments

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 the ixgbevf Intel Gigabit Linux driver, Copyright © 1999 - 2012 Intel Corporation, and distributed under the GPLv2 license, as published by the Free Software Foundation.

This product includes libwebp software. Copyright © 2010, Google Inc. All rights reserved.

This product includes Angular software developed by Google, Inc., <http://angularjs.org>, copyright © 2010-2012 Google, Inc., and distributed under the MIT license.

This product includes node.js software, copyright © Joyent, Inc. and other Node contributors. All rights reserved.

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 bootstrap software, copyright © 2011-2014 Twitter, Inc., and distributed under the MIT license (<http://getbootstrap.com/getting-started/#license-faqs>).

This product includes Intel PCM software, copyright © 2009-2013, Intel Corporation All rights reserved. This software is distributed under the OSI BSD license.

This product includes jxrlib software, copyright ©2009 Microsoft Corp. All rights reserved. Distributed under the new BSD license.

This product includes Net-SNMP software, to which one or more of the following copyrights apply:

- Copyright © 1989, 1991, 1992 by Carnegie Mellon University; Derivative Work - 1996, 1998-2000, Copyright © 1996, 1998-2000, The Regents of the University of California. All rights reserved. Distributed under CMU/UCD license (BSD like).
- Copyright © 2001-2003, Networks Associates Technology, Inc. All rights reserved. Distributed under the BSD license.
- Portions of this code are copyright © 2001-2003, Cambridge Broadband Ltd. All rights reserved. Distributed under the BSD license.
- Copyright © 2003 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved. Distributed under the BSD license.
- Copyright © 2003-2009, Sparta, Inc. All rights reserved. Distributed under the BSD license.
- Copyright © 2004, Cisco, Inc and Information Network Center of Beijing University of Posts and Telecommunications. All rights reserved. Distributed under the BSD license.
- Copyright © 2003 Fabasoft R&D Software GmbH & Co KG, oss@fabasoft.com. Distributed under the BSD license.
- Copyright © 2007 Apple Inc. All rights reserved. Distributed under the BSD license.
- Copyright © 2009 ScienceLogic, Inc. All rights reserved. Distributed under the BSD license.

This product includes Racoon 2 software, copyright © 2003-2005 WIDE Project. All rights reserved. Distributed under a BSD-like license.

This product includes node-uuid software, copyright © 2010-2012, Robert Kieffer, and distributed under the MIT license.

This product includes opensv software, which is distributed under the Apache 2.0 license.

Acknowledgments

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 iRules

- *What is an iRule?*
 - *Creating an iRule*
-

What is an iRule?

An *iRule* is a powerful and flexible feature within BIG-IP® Local Traffic Manager™ that you can use to manage your network traffic. Using syntax based on the industry-standard Tools Command Language (Tcl), the iRules® feature not only allows you to select pools based on header data, but also allows you to direct traffic by searching on any type of content data that you define. Thus, the iRules feature significantly enhances your ability to customize your content switching to suit your exact needs.

Important: For complete and detailed information on iRules syntax, see the F5 Networks DevCentral web site, <http://devcentral.f5.com>. Note that iRules must conform to standard Tcl grammar rules; therefore, for more information on Tcl syntax, see <http://tmm1.sourceforge.net/doc/tcl/index.html>.

An iRule is a script that you write if you want individual connections to target a pool other than the default pool defined for a virtual server. iRules allow you to more directly specify the destinations to which you want traffic to be directed. Using iRules, you can send traffic not only to pools, but also to individual pool members, ports, or URIs. The iRules you create can be simple or sophisticated, depending on your content-switching needs.

```
when CLIENT_ACCEPTED {
  if { [IP::addr [IP::client_addr] equals 10.10.10.10] } {
    pool my_pool
  }
}
```

This iRule is triggered when a client-side connection has been accepted, causing Local Traffic Manager to send the packet to the pool `my_pool`, if the client's address matches `10.10.10.10`.

Using a feature called the *Universal Inspection Engine*, you can write an iRule that searches either a header of a packet, or actual packet content, and then directs the packet based on the result of that search. iRules can also direct packets based on the result of a client authentication attempt.

iRules can direct traffic not only to specific pools, but also to individual pool members, including port numbers and URI paths, either to implement persistence or to meet specific load balancing requirements.

The syntax that you use to write iRules is based on the Tool Command Language (Tcl) programming standard. Thus, you can use many of the standard Tcl commands, plus a robust set of extensions that Local Traffic Manager provides to help you further increase load balancing efficiency.

Important: When referencing an object within an iRule, you must include the full path name of the object.

iRule commands

An *iRule command* within an iRule causes Local Traffic Manager™ to take some action, such as querying for data, manipulating data, or specifying a traffic destination. The types of commands that you can include within iRules® are:

Statement commands

These commands cause actions such as selecting a traffic destination or assigning a SNAT translation address. An example of a statement command is `pool <name>`, which directs traffic to the named load balancing pool.

Commands that query or manipulate data

Some commands search for header and content data, while others perform data manipulation such as inserting headers into HTTP requests. An example of a query command is `IP::remote_addr`, which searches for and returns the remote IP address of a connection. An example of a data manipulation command is `HTTP::header remove <name>`, which removes the last occurrence of the named header from a request or response.

Utility commands

These commands are functions that are useful for parsing and manipulating content. An example of a utility command is `decode_uri <string>`, which decodes the named string using HTTP URI encoding and returns the result.

Event declarations

iRules® are event-driven, which means that Local Traffic Manager™ triggers an iRule based on an event that you specify in the iRule. An *event declaration* is the specification of an event within an iRule that causes Local Traffic Manager to trigger that iRule whenever that event occurs. Examples of event declarations that can trigger an iRule are `HTTP_REQUEST`, which triggers an iRule whenever the system receives an HTTP request, and `CLIENT_ACCEPTED`, which triggers an iRule when a client has established a connection.

```
when HTTP_REQUEST {
  if { [HTTP::uri] contains "aol" } {
    pool aol_pool
  } else {
    pool all_pool
  }
}
```

Operators

An iRule operator compares two operands in an expression.

For example, you can use the `contains` operator to compare a variable operand to a constant. You do this by creating an `if` statement that represents the following: "If the HTTP URI contains `aol`, send to `pool aol_pool`."

Creating an iRule

You create an iRule® to customize the way that the BIG-IP® system processes traffic.

1. On the Main tab, click **Local Traffic > iRules**.
2. Click **Create**.
3. In the **Name** field, type a name, such as `my_irule`.
The full path name of the iRule cannot exceed 255 characters.
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**.

Chapter 2

iRule Commands

- *iRule command types*
- *The pool command*
- *The node command*
- *Commands that select a pool of cache servers*
- *The HTTP::*redirect* command*
- *The snat and snatpool commands*

iRule command types

There are three types of iRule commands:

- Statement commands
- Query and manipulation commands
- Utility commands (also known as functions)

Statement commands

Some of the commands available for use within iRules are known as statement commands. *Statement commands* enable Local Traffic Manager™ to perform a variety of different actions. For example, some of these commands specify the pools or servers to which you want Local Traffic Manager to direct traffic. Other commands specify translation addresses for implementing SNAT connections. Still others specify objects such as data groups or a persistence profiles.

For a complete list of statement commands, see the F5 Networks DevCentral web site, <http://devcentral.f5.com>.

Query and manipulation commands

Using iRules® commands, you can query for specific data contained in the header or content of a request or response, or you can manipulate that data. Data manipulation refers to inserting, replacing, and removing data, as well as setting certain values found in headers and cookies.

For example, using the `IP::idle_timeout` command within in iRule, you can query for the current idle timeout value that is set in a packet header and then load balance the packet accordingly. You can also use the `IP::idle_timeout` command to set the idle timeout to a specific value of your choice.

iRule query and manipulation commands are grouped into categories called *namespaces*. Except for commands in the global namespace, each iRule query or manipulation command includes the namespace in its command name. For example, one of the commands in the IP namespace is `IP::idle_timeout`. One of the commands in the HTTP namespace is `HTTP::header`.

For a complete list of namespaces for iRules commands, see the F5 Networks DevCentral web site, <http://devcentral.f5.com>.

Utility commands

Local Traffic Manager includes a number of utility commands that you can use within iRules. You can use these commands to parse and retrieve content, encode data into ASCII format, verify data integrity, and retrieve information about active pools and pool members.

The pool command

Once you have specified a query within your iRule, you can use the `pool` command to select a load balancing pool to which you want Local Traffic Manager™ to send a request. Here is an example of this command.

```
when HTTP_REQUEST {
  set uri [HTTP::uri]
  if { $uri ends_with ".gif" } {
    pool my_pool
  }
}
```



```

} elseif { $uri ends_with ".jpg" } {
    pool your_pool
}
}

```

The node command

As an alternative to the `pool` command, you can also write an iRule that directs traffic to a specific server. To do this, you use the `node` command.

```

when HTTP_REQUEST {
if { [HTTP::uri] ends_with ".gif" } {
    node 10.1.2.200 80
}
}

```

Commands that select a pool of cache servers

You can create an iRule that selects a server from a pool of cache servers. Here is an example of an iRule that selects a server from a pool of cache servers.

```

when HTTP_REQUEST
# This line specifies the expressions that determine whether the BIG-IP system sends requests
to the cache pool:
if { [HTTP::uri] ends_with "html" or [HTTP::uri] ends_with "gif" } {
    pool cache_pool
    set key [crc32 [concat [domain [HTTP::host] 2] [HTTP::uri]]]
    set cache_mbr [persist lookup hash $key node]
    if { $cache_mbr ne "" } {
        # This line verifies that the request is not coming from the cache:
        if { [IP::addr [IP::remote_addr] equals $cache_mbr] }
            # This line sends the request from the cache to the origin pool:
            pool origin_pool
            return
        }
    }
    # These lines ensure that the persistence record is added for this host/URI:
    persist hash $key
} else {
    pool origin_pool
}
}

```

Note: Local Traffic Manager™ redirects URIs to a new cache member at the time that the BIG-IP® system receives a request for the URI, rather than when the pool member becomes unavailable.

The HTTP::**redirect** command

In addition to configuring an iRule to select a specific pool, you can also configure an iRule to redirect an HTTP request to a specific location, using the `HTTP::redirect` iRule command. The location can be either a host name or a URI.

This is an iRule that is configured to redirect an HTTP response.

```
when HTTP_RESPONSE {
  if { [HTTP::status] contains "404" } {
    HTTP::redirect "http://www.siterequest.com/"
  }
}
```

Here is an example of an iRule that redirects an HTTP request.

```
when HTTP_REQUEST {
  if { [HTTP::uri] contains "secure" } {
    HTTP::redirect "https://[HTTP::host][HTTP::uri]"
  }
}
```

The **snat** and **snatpool** commands

The iRules[®] feature includes the two statement commands `snat` and `snatpool`. Using the `snat` command, you can assign a specified translation address to an original IP address from within the iRule, instead of using the SNAT screens within the BIG-IP Configuration utility.

Using the `snatpool` command also assigns a translation address to an original IP address, although unlike the `snat` command, the `snatpool` command causes Local Traffic Manager[™] to select the translation address from a specified SNAT pool that you previously created.

Chapter

3

iRule Evaluation

- *About iRule evaluation*
 - *Event types*
 - *iRule context*
 - *iRules assignment to a virtual server*
-

About iRule evaluation

In a basic system configuration where no iRule exists, Local Traffic Manager™ directs incoming traffic to the default pool assigned to the virtual server that receives that traffic. However, you might want Local Traffic Manager to direct certain kinds of connections to other destinations. The way to do this is to write an iRule that directs traffic to that other destination, contingent on a certain type of event occurring. Otherwise, traffic continues to go to the default pool assigned to the virtual server.

iRules® are therefore evaluated whenever an event occurs that you have specified in the iRule. For example, if an iRule includes the event declaration `CLIENT_ACCEPTED`, then the iRule is triggered whenever Local Traffic Manager accepts a client connection. Local Traffic Manager then follows the directions in the remainder of the iRule to determine the destination of the packet.

***Note:** When you make a change to an iRule with persistence, if there is already a connection in the connection table, the change does not take effect until the connection has expired. The same is true when you enable logging for the iRule and then change the iRule (or alter the logging message itself).*

Event types

The iRule command syntax includes several types of event declarations that you can specify within an iRule. For example:

- Global events, such as `CLIENT_ACCEPTED`
- HTTP events, such as `HTTP_REQUEST`
- SSL events, such as `CLIENTSSL_HANDSHAKE`
- Authentication events, such as `AUTH_SUCCESS`

For a complete list of iRule events and their descriptions, see the F5 Networks DevCentral web site, <http://devcentral.f5.com>.

iRule context

For every event that you specify within an iRule, you can also specify a context, denoted by the keywords `clientside` or `serverside`. Because each event has a default context associated with it, you need only declare a context if you want to change the context from the default.

The example shows `my_iRule1`, which includes the event declaration `CLIENT_ACCEPTED`, as well as the iRule command `IP::remote_addr`. In this case, the IP address that the iRule command returns is that of the client, because the default context of the event declaration `CLIENT_ACCEPTED` is `clientside`.

```
when CLIENT_ACCEPTED {
  if { [IP::addr [IP::remote_addr] equals 10.1.1.80] } {
    pool my_pool1
  }
}
```

Similarly, if you include the event declaration `SERVER_CONNECTED` in an iRule as well as the iRule command `IP::remote_addr`, the IP address that the iRule command returns is that of the server, because the default context of the event declaration `SERVER_CONNECTED` is `serverside`.

The preceding example shows what happens when you write an iRule that uses the default context when processing iRule commands. You can, however, explicitly specify the `clientside` and `serverside` keywords to alter the behavior of iRule commands.

Continuing with the previous example, the following example shows the event declaration `SERVER_CONNECTED` and explicitly specifies the `clientside` keyword for the iRule command `IP::remote_addr`. In this case, the IP address that the iRule command returns is that of the client, despite the server-side default context of the event declaration.

```
when SERVER_CONNECTED {
  if { [IP::addr [IP::addr [clientside {IP::remote_addr}] equals 10.1.1.80] ] {
    discard
  }
}
```

Note: You make an event declaration in an iRule by using the `when` keyword, followed by the event name. The figure shows an example of an event declaration in an iRule.

iRules assignment to a virtual server

When you assign multiple iRules® as resources for a virtual server, it is important to consider the order in which you list them on the virtual server. This is because Local Traffic Manager™ processes duplicate iRule events in the order that the applicable iRules are listed. An iRule event can therefore terminate the triggering of events, thus preventing Local Traffic Manager from triggering subsequent events.

Note: If an iRule references a profile, Local Traffic Manager processes this type of iRule last, regardless of its order in the list of iRules assigned to a virtual server.

Chapter

4

iRules and Administrative Partitions

- *iRules and administrative partitions*

iRules and administrative partitions

You should be aware of certain iRule configuration concepts as they relate to administrative partitions:

- An iRule can reference any object, regardless of the partition in which the referenced object resides. For example, an iRule that resides in `partition_a` can contain a pool statement that specifies a pool residing in `partition_b`.
- You can remove iRule assignments only from virtual servers that reside in the current `Write` partition or in partition `Common`.
- Note that you can associate an iRule only with virtual servers that reside in the current `Write` partition or in partition `Common`.
- You can associate an existing iRule with multiple virtual servers. In this case, the iRule becomes the only iRule that is associated with each virtual server in the current `Write` partition. Because this command overwrites all previous iRule assignments, F5 does not recommend use of this command.

Chapter 5

iRules and Local Traffic Profiles

- *iRules and profiles*
 - *The profile command*
 - *Commands that override profile settings*
-

iRules and profiles

When you are writing an iRule, you might want that iRule to recognize the value of a particular profile setting so that it can make a more-informed traffic management decision. Fortunately, the iRules® feature includes a command that is specifically designed to read the value of profile settings that you specify within the iRule.

Not only can iRules read the values of profile settings, but they can also override values for certain settings. This means that you can apply configuration values to individual connections that differ from the values Local Traffic Manager™ applies to most connections passing through a virtual server.

The profile command

The iRules® feature includes a command called `PROFILE`. When you specify the `PROFILE` command in an iRule and name a profile type and setting, the iRule reads the value of that particular profile setting. To do this, the iRule finds the named profile type that is assigned to the virtual server and reads the value of the setting that you specified in the `PROFILE` command sequence. The iRule can then use this information to manage traffic.

For example, you can specify the command `PROFILE::tcp idle_timeout` within your iRule. Local Traffic Manager™ then finds the TCP profile that is assigned to the virtual server (for example, `my_tcp`) and queries for the value that you assigned to the Idle Timeout setting.

Commands that override profile settings

Some of the iRule commands for querying and manipulating header and content data have equivalent settings within various profiles. When you use those commands in an iRule, and an event triggers that iRule, Local Traffic Manager™ overrides the values of those profile settings, using the value specified within the iRule instead.

For example, an HTTP profile might specify a certain buffer size to use for compressing HTTP data, but you might want to specify a different buffer size for a particular type of HTTP connection. In this case, you can include the command `HTTP::compress_buffer_size` in your iRule, specifying a different value than the value in the profile.

Chapter

6

Data Groups

- *About data groups*
-

About data groups

Data groups are useful when writing iRules®. A *data group* is simply a group of related elements, such as a set of IP addresses for AOL clients. When you specify a data group along with the `class match` command or the `contains` operator, you eliminate the need to list multiple values as arguments in an iRule expression.

You can define three types of data groups: address, integer, and string.

The BIG-IP® system includes three pre-configured data groups: `private_net`, `images`, and `aol`.

To understand the usefulness of data groups, it is helpful to first understand the `class match` command and the `contains` operator.

Note: You can manage only those data groups that you have permission to manage, based on your user role and partition access assignment.

Warning: Do not attempt to modify or delete any of the three pre-configured data groups (`private_net`, `images`, and `aol`). Doing so can produce adverse results.

About the class match command

The BIG-IP® system includes an iRule command called `class`, with a `match` option, which you can use to select a pool based on whether the command being used in the iRule represents a member of a specific data group. When you use the `class` command, the BIG-IP system knows that the string following the identifier is the name of a data group.

For example, using the `class` command, you can cause the BIG-IP system to load balance all incoming AOL connections to the pool `aol_pool`, if the value of the `IP::remote_addr` command is a member of the data group AOL. In this case, the class match command simply indicates that the object named `aol` is a collection of values (that is, a data group).

```
when CLIENT_ACCEPTED {
  if { [class match [IP::remote_addr] equals aol] } {
    pool aol_pool
  } else {
    pool all_pool
  }
}
```

Storage options

With Local Traffic Manager™, you can store data groups in two ways, either in-line or externally.

In-line storage

When you create data groups, Local Traffic Manager automatically saves them in their entirety in the `bigip.conf` file. This type of storage is known as *in-line storage*.

In general, in-line storage uses additional system resources due to extensive searching requirements on large data groups. For this reason, Local Traffic Manager offers you the ability to store your data groups externally, that is, outside of the `bigip.conf` file file.

External storage

You have the option to store data groups in another location on the BIG-IP® system, that is, outside of the `bigip.conf` file. Such data groups are called *external data groups*. Because the data group is stored externally in another location, the `bigip.conf` file itself contains only the filename and meta-data for the data group. The data in an externally-stored data group file is stored as a comma-separated list of values (CSV format).

Important: *If you attempt to load a `bigip.conf` file that contains external data group meta-data, and the file was created prior to BIG-IP system version 9.4, the system generates an error. The meta-data for the external data group contains the keyword `extern`, which generates an error during the load process. On BIG-IP systems running version 9.4 or later, the `extern` keyword is no longer needed in the `bigip.conf` file.*

To create an external data group, you first import a file from another location, using the **System** options of the BIG-IP Configuration utility. You then use the Local Traffic iRules® screens to create an external data group that is based on the imported file.

External data groups can scale to greater than 10,000,000 entries, depending on platform hardware and available memory (8 GB, or more, memory is recommended). Data groups with larger data items can be supported with fewer entries. Additionally, updates to external data groups are completely atomic: for example, the system updates a data group only after the new data successfully completes loading. You can use the command `[class exists xyz]` to check whether a data group has finished loading.

About file import for data groups

Using the BIG-IP® Configuration utility, you can import an external file that contains content that you want to use in a data group. When you import an existing file to the BIG-IP system, the BIG-IP system then creates a data group that contains the specified type of file content (address, string, or integer).

Importing a file for a data group

Using the BIG-IP Configuration utility, you can import a file from an external system and use the file to create a data group.

1. On the Main tab, click **System > File Management > Data Group File List > Import**.
2. For the **File Name** setting, click **Browse**.
The system opens a browse window so that you can locate the file that you want to import to the BIG-IP system.
3. In the **Name** field, type a new name for the imported file.
The new name appears in the list of imported files.
4. From the **File Contents** list, select the type of content for the data group.
5. In the **Key/Value Pair Separator** field, retain the default value or delete the value and specify a new separator.
6. In the **Data Group Name** field, type a name for the data group.
7. Click the **Import** button.

Viewing a list of imported data group files

Using the BIG-IP® Configuration utility, you can view a list of data group files that you have imported onto the BIG-IP system.

Data Groups

1. On the Main tab, click **System > File Management > Data Group File List**.
2. In the Name column, view the list of files.

Chapter 7

iFiles

- *About file import for iRules*
 - *About iFiles*
 - *iRule commands for referencing an iFile*
-

About file import for iRules

If you want to write an iRule that references a file that resides on another system, you must first import the file to the BIG-IP system. You can then convert the file to an iFile that an iRule can reference.

Importing a file for an iRule

Before you perform this task, the file you want to import must reside on the system you specify.

You can import a file from another system onto the BIG-IP® system, as the first step in writing an iRule that references that file.

1. On the Main tab, click **System > File Management > iFile List > Import**.
2. For the **File Name** setting, click **Browse**.
The system opens a browse window so that you can locate the file that you want to import to the BIG-IP system.
3. Browse for the file and click **Open**.
The name of the file you select appears in the **File Name** setting.
4. In the **Name** field, type a new name for the file, such as `lk.html`.
The new file name appears in the list of imported files.
5. Click the **Import** button.

After you perform this task, the file that you imported resides on the BIG-IP system.

Viewing a list of imported files

You perform this task to view a list of files that you have imported to the BIG-IP® system for the purpose of referencing the files within iRules.

1. On the Main tab, click **System > File Management > iFile List**.
2. In the Name column, view the list of iFiles.
3. Click the **Cancel** button.

About iFiles

Using the BIG-IP Configuration utility, you can create a special file called an iFile. An *iFile* is a file that is based on an external file that you previously imported to the BIG-IP® system from another system. You can reference an iFile from within an iRule, based on a specific iRule event.

To create an iFile and use it within an iRule, you start from the **Local Traffic** option on the Main tab.

Important: Prior to creating an iFile, you must import a file to the BIG-IP system from another system.

Viewing a list of iFiles

You perform this task to view a list of files that you have imported to the BIG-IP® system for the purpose of referencing the files within iRules.

1. On the Main tab, click **Local Traffic > iRules > iFile List**.
2. In the Name column, view the list of iFiles that you previously created on the BIG-IP system.

Creating an iFile

As a prerequisite, ensure that the current administrative partition is set to the partition in which you want the iFile to reside. Also ensure that the file has been imported to the BIG-IP® system.

You perform this task to create an iFile that you can then reference in an iRule.

1. On the Main tab, click **Local Traffic > iRules > iFile List**.
2. Click **Create**.
3. In the **Name** field, type a new name for the iFile, such as `ifileURL`.
4. From the **File Name** list, select the name of the imported file object, such as `1k.html`.
5. Click **Finished**.
The new iFile appears in the list of iFiles.

The result of this task is that you now have a file that an iRule can reference.

iRule commands for referencing an iFile

With these iRule commands, you can reference the new iFile from within an iRule:

- `[ifile get IFILENAME]`
- `[ifile listall]`
- `[ifile attributes IFILENAME]`
- `[ifile size IFILENAME]`
- `[ifile last_updated_by IFILENAME]`
- `[ifile last_update_time IFILENAME]`
- `[ifile revision IFILENAME]`
- `[ifile checksum IFILENAME]`
- `array set [file attributes IFILENAME]`

This sample iRule shows some of these commands:

```
ltm rule ifile_rule {
  when HTTP_RESPONSE {
    # return a list of iFiles in all partitions
    set listifiles [ifile listall]
    log local0. "list of ifiles: $listifiles"

    # return the attributes of an iFile specified
    array set array_attributes [ifile attributes "/Common/ifileURL"]
    foreach {array_attr} [array get array_attributes ] {
      log local0. "$array : $attr"
    }
  }
}
```

```
}  
  
# serve an iFile when http status is 404.  
set file [ifile get "Common/ifileURL"]  
log local0. "file: $ifile"  
if { [HTTP::status] equals "404" } {  
    HTTP::respond 200 ifile "/Common/ifileURL"  
}  
}  
}
```

Index

A

administrative partitions
and iRules 32

C

cache server pools 25
class match command 36

D

data group files
 address 37
 example 37
 importing 37
 integer 37
 string 37
 viewing list of 37
data groups
 defined 36
 storage of 36

E

event declarations
 defined 21

F

file import
 for iRules 40
files
 importing 40–41

H

HTTP::redirect command 26

I

iFile commands 41
iFile files
 defined 40
iFiles
 creating 41
imported files
 listing 40–41
iRule commands
 types of 20, 24
iRule context 28

iRule evaluation 28
iRule event types 28
iRule operators
 defined 21
iRules
 and iFiles 41
 and profiles 34
 and virtual servers 29
 creating 21
 described 20
 importing files for 40
 ordering of 29

N

node command 25

O

operators
 defined 21

P

partitions
 and iRules 32
pool command 24
profile command 34
profiles
 and iRules 34
profile settings
 overriding with iRules 34

S

snat and snatpool commands 26
storage
 of data groups 36

T

Tcl
 and iRules 20
Tools Command Language
 and iRules 20

V

virtual servers
 and iRules 29

