IBM

Implementation of IBM j-type Ethernet Switches and Routers

- Introduction to Ethernet fundamentals and IBM j-type Ethernet switches and routers
- Introduction to Junos software and the command-line interface
- Guidance for planning, installation, and configuration

Sangam Racherla
Norman Bogard
Gareth Edwards
Nathan Flowers
Paul Ionescu
Gabriel Slomovitz
See Keong Soon

Redbooks

ibm.com/redbooks

International Technical Support Organization

Implementation of IBM j-type Ethernet Switches and Routers

February 2011

SG24-7882-00

Note: Before using this information and the product it supports, read the information in “Notices” on page xi.

First Edition (February 2011)

This edition applies to IBM j-type Ethernet switches and routers operating on Junos Software Version 10.1.

© Copyright International Business Machines Corporation 2011. All rights reserved.
Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices . . . xi
Trademarks . . . xii

Preface . . . xiii
The team who wrote this book . . . xiv
Now you can become a published author, too! . . . xvi
Comments welcome . . . xvi
Stay connected to IBM Redbooks . . . xvii

Chapter 1. Fundamentals of Ethernet networking . . . 1
1.1 Open System Interconnect model . . . 2
1.2 General networking concepts . . . 3
1.2.1 Local area network . . . 3
1.2.2 Wide area network . . . 4
1.3 Layer 1 networking concepts and terminology . . . 5
1.3.1 Ethernet cabling . . . 5
1.3.2 Components . . . 12
1.3.3 Network segments . . . 12
1.3.4 Physical configuration parameters . . . 13
1.3.5 Power over Ethernet . . . 14
1.4 Layer 2 networking concepts and terminology . . . 15
1.4.1 Frame structure . . . 16
1.4.2 Jumbo frames . . . 17
1.4.3 Interframe gap . . . 17
1.4.4 MAC addresses . . . 17
1.4.5 Bridging or Layer 2 switching . . . 19
1.4.6 Virtual local area network . . . 23
1.4.7 Interface VLAN operation modes . . . 25
1.4.8 Link aggregation . . . 26
1.4.9 Spanning Tree Protocol . . . 26
1.4.10 Link Layer Discovery Protocol . . . 29
1.4.11 LLDP TLVs . . . 29
1.5 Layer 3 networking concepts and terminology . . . 31
1.5.1 IP routing . . . 32
1.5.2 Address Resolution Protocol . . . 33
1.5.3 IPv4 addressing . . . 34
1.5.4 IPv6 addressing . . . 36

Chapter 2. Introduction to IBM j-type Ethernet switches and routers . . . 39
2.1 IBM j-type e-series Ethernet switches . . . 40
2.1.1 IBM Ethernet Switch J48E . . . 41
2.1.2 IBM Ethernet Switch J08E and J16E . . . 44
2.2 IBM j-type m-series Ethernet routers . . . 51
2.2.1 Key components of IBM j-type m-series Ethernet routers . . . 52
2.3 Junos operating system . . . 61
2.4 Simplification of the IBM Data Center Network solution . . . 63
2.5 More information . . . 65

Chapter 3. Initial hardware planning of IBM j-type Ethernet switches and routers . . . 67

© Copyright IBM Corp. 2011. All rights reserved.

3.1 Power considerations 68
3.1.1 Power specifications and requirements for IBM Ethernet Switch J48E 68
3.1.2 Power specifications and requirements for IBM Ethernet Switches J08E and J16E 70
3.1.3 Power specifications and requirements for IBM Ethernet Routers J02M, J06M, and J11M 75
3.2 Cooling system 83
3.2.1 IBM Ethernet Switch J48E 83
3.2.2 IBM Ethernet Switches J08E and J16E 85
3.2.3 IBM Ethernet Routers J02M, J06M, and J11M 87
3.3 Racks 91
3.3.1 IBM Ethernet Switch J48E 91
3.3.2 IBM Ethernet Switches J08E and J16E 94
3.3.3 IBM Ethernet Routers J02M, J06M, and J11M 96
3.4 Cabling 99
3.4.1 Cables connecting to management devices 99
3.5 More information 105

Chapter 4. Fundamental concepts of the Junos operating system 107
4.1 Overview of Junos 108
4.2 The Junos architecture 109
4.3 Routing Engine 109
4.3.1 Routing Engine components for Junos software 109
4.4 Packet Forwarding Engine 111
4.5 The Junos software configuration model 111
4.5.1 Active configuration 111
4.5.2 Candidate configuration 111
4.5.3 Configuration history 112
4.6 Junos routing and forwarding tables 112
4.7 User interface options 113
4.8 Introduction to Junos CLI 114
4.8.1 Operational mode 114
4.8.2 Configuration mode 114
4.8.3 More information 115

Chapter 5. Initial configuration 117
5.1 Network topology 118
5.2 Initial setup for the J08E, J16E, and J48E switches 119
5.2.1 Configuring the switch from the console 122
5.2.2 Configuring the switch through the J-Web interface 124
5.3 LCD panel and status LEDs 133
5.3.1 LCD panel 133
5.3.2 Chassis status LEDs 137
5.4 Initial setup for the J02M, J06M, and J11M routers 139
5.4.1 Initial configuration 139

Chapter 6. User interface 143
6.1 J-Web GUI 144
6.1.1 Logging in to the J-Web GUI 144
6.1.2 Dashboard tab 148
6.1.3 Configure tab 149
6.1.4 Monitor tab 150
6.1.5 Maintain tab 151
6.1.6 Troubleshoot tab 152

6.2 Junos CLI 152
6.2.1 CLI Modes. 152
6.2.2 Logging in CLI. 153
6.2.3 CLI operational mode 154
6.2.4 CLI configuration mode. 160
6.3 More information 168

Chapter 7. Virtual Chassis 169
7.1 Overview of a Virtual Chassis 170
7.1.1 Benefits of a Virtual Chassis 170
7.1.2 Components of a Virtual Chassis 170
7.1.3 Cabling options for a Virtual Chassis 178
7.2 Deployment options for a Virtual Chassis 180
7.2.1 Top-of-rack deployment in a data center 180
7.2.2 End-of-row deployment in a data center 184
7.3 Installing a Virtual Chassis configuration 184
7.3.1 Building a new Virtual Chassis with the dynamic method. 185
7.3.2 Building a new Virtual Chassis with the preprovisioning method 188
7.3.3 Adding a new switch to an existing Virtual Chassis configuration in the same telecommunications closet 192
7.3.4 Adding a new switch from a remote telecommunications closet to an existing Virtual Chassis configuration 194
7.3.5 Replacing a member switch 196
7.4 Management of a Virtual Chassis 200
7.4.1 Console port access 200
7.4.2 Management Ethernet port access 201
7.4.3 Management IP address access. 202
7.4.4 Synchronizing the configuration to Virtual Chassis members. 202
7.5 Operational monitoring of a Virtual Chassis 203
7.5.1 Verifying the member ID, status, and role of a Virtual Chassis member. 203
7.5.2 Verifying the status of a Virtual Chassis port 204
7.6 More information 205

Chapter 8. Layer 1: Configuration 207
8.1 Network topology for Layer 1 configuration. 208
8.2 Configuration of the port settings 208
8.2.1 Configuring the link settings on IBM j-type Ethernet switches 208
8.2.2 Configuring the link settings on IBM j-type Ethernet routers. 214
8.2.3 Common configuration options for interfaces 215
8.3 Configuring PoE 217
8.3.1 Configuring PoE on an IBM Ethernet Switch J48E. 217
8.3.2 Verifying the status of PoE interfaces on an IBM Ethernet Switch J48E 220
8.4 More information 221

Chapter 9. Layer 2: Switching configuration 223
9.1 Network topology for Layer 2 configuration. 224
9.2 Link aggregation configuration 225
9.2.1 Link aggregation group platform characteristics 225
9.2.2 Configuring link aggregation 225
9.2.3 Verifying link aggregation 227
9.2.4 A sample LAG configuration 227
9.2.5 Configuring link aggregation with LACP 232
9.2.6 Sample LAG with LACP configuration 233
9.3 Configuring a VLAN 237

9.4 Configuring a port VLAN . . . 239
9.4.1 Access ports . . . 239
9.4.2 Trunk ports . . . 242
9.4.3 Routed VLAN interface . . . 245
9.5 Configuring the Spanning Tree Protocol . . . 247
9.5.1 Spanning Tree Protocol . . . 248
9.5.2 Rapid Spanning Tree Protocol . . . 249
9.5.3 Multiple Spanning Tree Protocol . . . 252
9.5.4 VLAN Spanning Tree Protocol . . . 256
9.5.5 BPDU protection . . . 259
9.5.6 Loop protection . . . 260
9.5.7 Root protection . . . 261
9.6 Link Layer Discovery Protocol . . . 261
9.6.1 Understanding LLDP . . . 261
9.6.2 LLDP TLVs . . . 262
9.6.3 Considerations for LLDP and LLDP-MED . . . 263
9.6.4 Configuring and monitoring LLDP . . . 263
9.7 More information . . . 270

Chapter 10. Layer 3: Routing configuration . . . 273
10.1 Network topology for Layer 3 configuration . . . 274
10.2 IP routing . . . 275
10.2.1 Routing protocol databases . . . 275
10.2.2 Junos routing tables . . . 276
10.2.3 Forwarding tables . . . 276
10.2.4 Synchronizing the routing and forwarding tables . . . 277
10.2.5 Route preferences overview . . . 277
10.2.6 Multiple active routes . . . 277
10.2.7 Determining the active route . . . 277
10.2.8 Default route preference values . . . 278
10.3 Routing protocols . . . 278
10.3.1 RIP . . . 278
10.3.2 OSPF . . . 287
10.3.3 BGP . . . 301
10.4 Static routes . . . 317
10.5 Multicast . . . 320
10.6 Virtual Router Redundancy Protocol . . . 324
10.7 More information . . . 330

Chapter 11. Class of service . . . 333
11.1 Overview of class of service . . . 334
11.2 Junos class-of-service components . . . 334
11.3 Hardware limitations . . . 336
11.4 Packet flow . . . 336
11.5 Packet classification . . . 337
11.5.1 Behavior aggregate classification . . . 337
11.5.2 Code point alias . . . 341
11.5.3 Multifield classification . . . 343
11.6 Forwarding classes . . . 345
11.7 Policing . . . 346
11.7.1 Two-color marking configuration . . . 347
11.7.2 Two-color marking example . . . 349
11.8 Rewrite rules . . . 349

11.8.1 Default rewrite rules 350
11.8.2 Configuring rewrite rules 351
11.8.3 Example of a default rewrite rule. 351
11.9 Packet loss priority 352
11.10 Schedulers 352
11.10.1 Default schedulers 352
11.10.2 Configuring schedulers 353
11.10.3 Example of a scheduler configuration. 355
11.11 RED profiles 356
11.11.1 Configuration. 356
11.11.2 Example 357
11.12 Tail drop profiles 360
11.12.1 Configuration. 360
11.12.2 Example 361
11.13 Case study 361
11.14 More information 366

Chapter 12. Security 367
12.1 Access security 368
12.1.1 Access overview 368
12.1.2 Securing access to switches and routers 368
12.2 Firewall filters 371
12.2.1 Firewall filter types 372
12.2.2 Firewall filter components 373
12.2.3 Firewall filter processing 373
12.2.4 Examples 374
12.3 Port security 381
12.3.1 Features of port security 381
12.3.2 MAC limiting and MAC move limiting 382
12.3.3 Port security features in a Layer 2 network. 385
12.4 802.1X authentication 390
12.4.1 Overview of 802.1X authentication 390
12.4.2 Example: Configuring 802.1X authentication 393
12.5 More information 408

Chapter 13. System management 409
13.1 Overview of network management 410
13.1.1 Understanding device management functions in Junos software. 410
13.2 Overview of the SNMP 411
13.2.1 Architecture of the SNMP 412
13.2.2 Features of the Junos SNMP agent 412
13.2.3 Management Information Base. 413
13.2.4 SNMP traps and informs. 413
13.3 Configuring SNMP 414
13.3.1 Configuring SNMP on a device running Junos software. 415
13.3.2 Configuring the system contact on a device running Junos software 415
13.3.3 Configuring the system location for a device running Junos software 415
13.3.4 Configuring the description on a device running Junos software 415
13.3.5 Filtering duplicate SNMP requests 416
13.3.6 Configuring the commit delay timer. 416
13.3.7 Configuring the system name 417
13.3.8 Configuring the SNMP community string and options. 417
13.3.9 Configuring the SNMP trap options. 417

13.3.10 Configuring SNMP trap groups . . . 418
13.3.11 Configuring MIB views . . . 420
13.3.12 Tracing SNMP activity on a device running Junos software . . . 421
13.3.13 Configuring the SNMP log file . . . 421
13.4 Overview and configuration of SNMPv3 . . . 423
13.4.1 SNMPv3 configuration statements . . . 424
13.4.2 Minimum SNMPv3 configuration on a device running Junos software . . . 426
13.4.3 Configuring the local engine ID . . . 428
13.4.4 Creating SNMPv3 users . . . 428
13.4.5 Configuring the SNMPv3 authentication type . . . 429
13.4.6 Configuring the encryption type . . . 430
13.4.7 Example: Defining SNMPv3 users . . . 431
13.4.8 Defining access privileges for an SNMP group . . . 434
13.4.9 Configuring the access privileges granted to a group . . . 435
13.4.10 Example: Defining access privileges . . . 437
13.4.11 Assigning security names to groups . . . 438
13.4.12 Configuring SNMPv3 traps on a device running Junos software . . . 440
13.4.13 Configuring the SNMPv3 trap notification . . . 440
13.4.14 Configuring the trap notification filter . . . 441
13.4.15 Configuring the trap target address . . . 442
13.4.16 Example: Configuring the tag list . . . 444
13.4.17 Defining and configuring the trap target parameters . . . 445
13.4.18 Configuring SNMP informs . . . 447
13.4.19 Configuring the remote engine and remote user . . . 448
13.4.20 Configuring the inform notification type and target address . . . 449
13.4.21 Configuring the SNMPv3 community . . . 450
13.5 MIB support in Junos software . . . 451
13.5.1 Standard SNMP MIBs supported by Junos software . . . 451
13.5.2 Loading MIB files to a network management station . . . 457
13.6 Using sFlow to monitor on a j-type e-series Ethernet switch . . . 458
13.6.1 sFlow components . . . 460
13.6.2 Configuring sFlow technology . . . 461
13.6.3 Checking the results of a configuration . . . 461
13.6.4 Summary of the sFlow configuration . . . 462
13.6.5 Viewing the collected data . . . 462
13.7 Configuring system log messages . . . 464
13.7.1 Minimum and default system logging configuration . . . 464
13.7.2 Displaying and interpreting system log messages . . . 480
13.8 More information . . . 489

Chapter 14. Maintenance and analysis . . . 491
14.1 Basic systems functions . . . 492
14.1.1 Rebooting or halting a system . . . 492
14.1.2 Managing files . . . 494
14.1.3 Setting a rescue configuration . . . 495
14.1.4 Reverting to the rescue configuration . . . 495
14.1.5 Reverting to the default factory settings . . . 496
14.1.6 Recovering the root password . . . 497
14.2 Network utilities . . . 498
14.2.1 The ping and traceroute commands . . . 498
14.2.2 Port mirroring on IBM j-type e-series Ethernet switches . . . 499
14.2.3 Port mirroring on IBM j-type m-series Ethernet routers . . . 503
14.2.4 Telnet, SSH, and FTP clients . . . 503

14.3 Tools for diagnosing the device 504
14.3.1 The show interfaces command 504
14.3.2 The monitor interface command 508
14.3.3 The monitor traffic command 509
14.3.4 Monitoring system commands 511
14.3.5 Viewing log and trace files 512
14.4 Managing the Junos software 512
14.4.1 Software packaging 513
14.4.2 System snapshot 513
14.4.3 Upgrading the Junos software 514
14.5 Overview of the file system 521
14.6 Managing the configuration 523
14.6.1 Saving the configuration 523
14.6.2 Returning to the most recently committed configuration 524
14.6.3 Returning to a previously committed configuration 524
14.6.4 Viewing previous configurations 524
14.6.5 Comparing configuration changes 525
14.6.6 Saving a configuration to a file 526
14.6.7 Loading a configuration from a file 526
14.7 Chassis and interfaces alarms 528
14.7.1 Alarm relay contacts 528
14.7.2 Craft interface LEDs 528
14.7.3 Component LEDs 529
14.8 More information 530

Appendix A. Preferred practices for networking 531
Monitoring and management 532
Traffic isolation and reduction 533
Cabling 533
Port configuration 534
Additional resources 535

Related publications 537
IBM Redbooks 537
Other publications 537
Online resources 539
Help from IBM 539

Index 541

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

© Copyright IBM Corp. 2011. All rights reserved.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®	POWER®	System p®
BladeCenter®	Redbooks®	System x®
IBM®	Redbooks (logo) ®	

The following terms are trademarks of other companies:

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Preface

The data center is the strategic heart of today's high-performance enterprise. It unifies the critical systems, applications, and storage services that are needed for business success. With a data center solution designed by IBM® and Juniper Networks, companies can centralize their infrastructure with a comprehensive solution that combines best-in-class products with well-defined practices for the enterprise.

IBM j-type data center solutions running Junos software provide operational agility and efficiency, dramatically simplifying the network and delivering unprecedented savings. With this solution, a network design has fewer devices, interconnections, and network tiers. Beyond the obvious cost advantages, the design offers the following key benefits:

- Reduces latency
- Simplifies device management
- Delivers significant power, cooling, and space savings
- Eliminates multiple system failure points
- Performs pervasive security

The high-performance data center is built around IBM j-type e-series Ethernet switches, m-series routers, and s-series firewalls. This new family of powerful products helps to shape the next generation of dynamic infrastructure.

IBM j-type e-series Ethernet switches meet escalating demands while controlling costs. They provide operational simplicity and efficiency for interconnecting data centers and enabling optimal communications between servers, storage, and users.

IBM j-type m-series Ethernet routers are high-performance routers with powerful switching and security capabilities. They are designed for deployments in the data center, core and aggregation configurations, and WAN Edge configurations. They have advanced routing features such as Multiprotocol Label Switching (MPLS) network virtualization and low-latency multicast.

This IBM Redbooks® publication targets IT professionals who sell, design, or administer IBM j-type networking solutions. It provides information about IBM j-type Ethernet switches and routers and includes the following major topics:

- Introduction to Ethernet fundamentals and IBM j-type Ethernet switches and routers
- Initial hardware planning and configuration
- Other configuration topics including Virtual Chassis configuration, Layer 1, Layer 2, and Layer 3 configurations, and security features
- Network management features of Junos software and maintenance of the IBM j-type series hardware

This Redbooks publication can be used with the following publications:

- *IBM j-type Data Center Networking Introduction*, SG24-7820
- *IBM j-type Ethernet Appliance Implementation*, SG24-7883

© Copyright IBM Corp. 2011. All rights reserved.

The team who wrote this book

This book was produced by a team of specialists from around the world working at the International Technical Support Organization (ITSO), Raleigh Center.

Sangam Racherla is an IT Specialist and Project Leader working at the ITSO in San Jose, CA. He has 10 years of experience in the IT field and has been with the ITSO for the past seven years. Sangam has extensive experience in installing and supporting the ITSO lab equipment for various Redbooks projects. He has expertise in working with Microsoft® Windows®, Linux®, IBM AIX®, System x®, and System p® servers, and various SAN and storage products. Sangam holds a degree in electronics and communication engineering.

Norman Bogard is a Senior Technical Sales Specialist with IBM Advanced Technical Skills Organization. He has been supporting Ethernet networking since 1987, storage area networks (SAN) since 1996, and network-attached storage (NAS) since IBM's entry into the sector in 2001. He began his career in IT in 1984 with Intel® Corporation and came into IBM through the acquisition of Sequent Computers.

Gareth Edwards is an IT Specialist for STG in IBM Australia specializing in Storage and IBM POWER® systems. He has 20 years of experience in the IT industry and has been with IBM since 2006. Gareth's current area of expertise includes the planning and implementation of midrange and enterprise servers, storage, and storage networks. He is certified on IBM Midrange Storage Systems and Cisco Networking.

Nathan Flowers is a Network Engineer and has been with IBM for 20 years. He started his career in the National Support Center in Atlanta, GA, providing dealer support for PC, XT, AT, PS/2s, and Thinkpad systems. After moving to IBM in Raleigh, he joined the development group of the Network Hardware Division as an Asynchronous Transfer Mode (ATM) and Ethernet network engineer in 1998. Nathan has had many roles in the networking environment including IBM BladeCenter® networking solutions and BladeCenter and System x networking education. He is currently in the SAN Central and Solutions support group, where he provides product engineering and solutions support for IBM Data Center Networking products. He holds a Bachelor of Science degree in computer science from Kennesaw State University.

Paul Ionescu is a Senior IT Specialist working for IBM Global Technology Services Romania with more than 10 years of networking experience. Paul joined IBM in 2000. He is currently focused on network engineering projects for the Internet Service Provider (ISP), telecommunications, and banking industries. He is a Juniper Networks Certified Internet Expert (JNCIE-M), Cisco Certified Internetworking Expert (CCIE), and Certified Information Systems Security Professional (CISSP).

Gabriel Slomovitz is a Network Specialist at IBM Uruguay. He has almost 10 years of experience working in design and implementation of networking projects. His specialties include routing, switching, wireless, and RFID. He is a Cisco Certified Internetwork Expert (CCIE) in routing and switching and holds an engineering degree in telecommunications from Universidad ORT.

See Keong Soon is an Advisory IT Specialist for IBM Global Technology Services in Malaysia. He has eight years of experience and specializes in network and security technologies. Among many previous appointments he has held, See Keong served as a network lead to implement many large-scale network and security projects. His areas of expertise include planning, designing, and implementation of various network and security technologies for a wide range of clients.

The team (from left to right): Gareth Edwards, Gabriel Slomovitz, See Keong Soon, Norman Bogard, Paul Ionescu, Nathan Flowers, and Sangam Racherla

Thanks to the following people for their contributions to this project:

Tamikia Barrow
Jon Tate
Margaret Ticknor
ITSO, Raleigh Center

Mark Bayus
Jim Blue
William Champion
Jason Daniel
Steve Grillo
Doris Konieczny
George Pappas
Doug Vassello
Jeffrey Walls
IBM US

Greg Basset
Sean Capshaw
Yue Chen
Vaishali Ghiya
Steve Gonzales
Charles Goldberg
John Nishikawa
Sara Riley
Chris Rogers
Mark Rosche
Steven Smith

Fraser Street
Jeremy Wallace
Juniper Networks

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author - all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks

- Follow us on Twitter:

  http://twitter.com/ibmredbooks

- Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm

- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

1

Fundamentals of Ethernet networking

This chapter explains the fundamental networking concepts and terminology that is related to topics covered in this book. For the scope of this book, this chapter provides information about the following topics:

- Open System Interconnect model
- General networking concepts
- Layer 1 networking concepts and terminology
- Layer 2 networking concepts and terminology
- Layer 3 networking concepts and terminology

© Copyright IBM Corp. 2011. All rights reserved.

1.1 Open System Interconnect model

The Open System Interconnect (OSI) reference model divides the computer networking architecture into to seven layers as illustrated in Figure 1-1. It provides a visual representation of the dependencies and interactions of various communication processes. Processes are grouped according to their function in the model. Processes at each layer depend upon the functions of the layer below and provides services for the layer above.

Figure 1-1 OSI layers

The OSI model has the following layers. Each description includes examples of the layer's functions.

Layer 7, Application Provides access to network processes to and from applications, for example: FTP, SMTP, HTTP, SNMP, Telnet, and NTP.

Layer 6, Presentation Provides data representation and encryption, for example: Secure Sockets Layer (SSL), Transport Layer Security (TLS), and Extensible Markup Language (XML).

Layer 5, Session Provides interhost communications or process-to-process messages between computers. Manages establishment and termination of sessions. Examples include socket addresses.

Layer 4, Transport Provides end-to-end communications and reliability. Manages flow control, fragmentation, and error control. Examples include Transmission Control Protocol (TCP) functions of TCP/IP.

Layer 3, Network Provides path determination or routing from source to destination, for example, Internet Protocol (IP) functions of TCP/IP.

Layer 2, Data Link Provides the functional means for data transfer between adjacent nodes in the network, for example, Media Access Control (MAC) or physical addressing.

Layer 1, Physical Provides electrical and physical parameters (signal and media) for transmission of data, for example: functions of Network Interface Cards (NICs), hubs, cables, connector types, and signaling parameters.

1.2 General networking concepts

Networks can be categorized by many factors. Two of the most common terms are local area network (LAN) and wide area network (WAN).

1.2.1 Local area network

A LAN is a computer network of devices that spans a small geographic area. A LAN can be a small network within a small office or home. A LAN can also be a large network such as in a large office building or campus. The boundary of the LAN is commonly the point at which the traffic exits the management control of the LAN and enters the management control of an Internet Service Provider (ISP).

1.2.2 Wide area network

A WAN interconnects multiple LANs. The area of a WAN can vary from interconnecting LANs that are near each other or can span a region, country, or the world as illustrated in Figure 1-2. A WAN can be composed of many technologies with a wide variety of capabilities and features.

Figure 1-2 A WAN interconnecting LANs

1.3 Layer 1 networking concepts and terminology

Layer 1 of the OSI model is the layer at which the physical transmission of data occurs. The unit of transmission at Layer 1 is a *bit*. This section explains some of the common concepts that are at the Layer 1 level.

1.3.1 Ethernet cabling

Ethernet cabling is one of two forms: copper cabling or fiber optic cabling. Copper is typically the least expensive choice for materials, components, and installation. Copper cabling is the method commonly used to connect devices to the access layer switches.

Fiber optic cabling is more expensive than copper cabling. The optical components for devices and switches and the cost of any customer cabling is typically more expensive to install. However, the higher costs are often easily justified by the benefits of fiber optic cabling.

Fiber optic cabling provides for longer distance and is resistant to the signaling being distorted by electromagnetic interference.

Copper cabling

An Ethernet uses two forms of copper cabling: coaxial and twisted pair.

Coaxial cabling

The early Ethernet used coaxial copper cabling in two sizes:

- 10BASE2 (RG58/RG6) “Thinnet” for distances up to 185 meters
- 10BASE5 (RG8/11) “Thicknet” for distances up to 500 meters

For coaxial cabling, the common connector types used are BNC (Figure 1-3) and Type F (Figure 1-4).

Figure 1-3 BNC Connector

Figure 1-4 Type F Connector

Twisted-pair cabling

Twisted-pair copper cabling is a common media for Ethernet networking installations. Twisted-pair cabling is available as Unshielded Twisted-Pair (UTP) or Shielded Twisted-Pair (STP). This shielding helps prevent electromagnetic interference.

Several different categories of twisted-pair cabling are available as listed in Table 1-1. These categories indicate the signaling capabilities of the cabling.

Table 1-1 TIA/EIA cabling categories

TIA/EIA cabling category	**Maximum network speeds supported**
Cat 1	Telephone or ISDN
Cat 2	4 Mb Token Ring
Cat 3	10 Mb Ethernet
Cat 4	16 Mb Token Ring
Cat 5	100 Mb Ethernet
Cat 5e	1 Gb Ethernet
Cat 6	10 Gb Ethernet Short Distance - 55 m (180 ft.)
Cat 6a	10 Gb Ethernet

The connector used for Ethernet twisted-pair cabling is likely the one most people recognize and associate with networking, the RJ45 connector, which is shown in Figure 1-5.

Figure 1-5 RJ45 Copper Connector

Twisted-pair cabling contains four pairs of wire inside the cable, as illustrated in Figure 1-6.

Figure 1-6 Straight through Ethernet cable

An Ethernet operating in 10/100 Mb mode only uses two pairs, pairs 1-2 and 3-6. An Ethernet operating in 1 Gb mode uses all four pairs: pairs 1-2, 3-6, 4-5, and 7-8. Distances up to 100 meters are supported.

Damaged twisted pair: If a twisted-pair cable is damaged, so that pair 4-5 or pair 7-8 is unable to communicate, the link is unable to communicate in 1 Gbps mode. If the devices are set to autonegotiate speed, the devices successfully operate in 100 Mbps mode.

Supported maximum distances of cabling segment: The actual maximum distances of a cabling segment that are supported vary on multiple factors such as vendor support, cabling type, electromagnetic interference, and number physical connections in the segment.

Twisted-pair crossover requirements

In 10/100 Mbps Ethernet operations, one pair of wire is used for data transmission and one pair is used for receiving data. When a device, such as a PC, is attached to a hub or switch, the ports are designed so that the transmitting and receiving pairs are properly matched. When directly connecting two like devices, such PC-PC, hub-hub, or switch-switch, a crossover in the pairs must be made.

A crossover function can be made internally by the port of one of the devices or can be achieved by using a crossover cable as illustrated in Figure 1-7.

Figure 1-7 10/100 Mbps crossover cable

Ethernet ports without crossover are known as *Medium Dependent Interface* (MDI). Ports with crossover are known as *Medium Dependent Interface Crossover* (MDIX). The “X” means crossover. Some ports can sense whether crossover is needed and configure the port properly. This function is often referred to as *Auto MDIX*. For Gigabit Ethernet, the auto crossover function is an optional part of the 1000 Base-T Ethernet standard.

Fiber optic cabling

In copper cabling, electric signals are used to transmit data through the network. The copper cabling was the medium for that electrical transmission. In fiber optic cabling, light is used to transmit the data. Fiber optic cabling is the medium for channeling the light signals between devices in the network.

Two modes of fiber optic signaling are explained in this chapter, single-mode and multimode. The difference between the modes is the wavelength of the light used for the transmission as illustrated in Figure 1-8.

Figure 1-8 Multimode versus single-mode optic signaling

Single-mode fiber

Single-mode fiber (SMF) uses long wavelength light to transmit data and requires a cable with a small core for transmission (Figure 1-8 on page 8). The core diameter for single-mode cabling is 9 microns in diameter (Figure 1-9).

Figure 1-9 Single-mode fiber cable

Multimode fiber

Multimode fiber (MMF) uses short wavelength light to transmit data and requires a cable with a larger core for transmission (Figure 1-8 on page 8). The core diameter for multimode cabling can be 50 or 62.5 microns in diameter as illustrated in Figure 1-10.

Figure 1-10 Multimode fiber cable

The color of the outer coating is sometimes used to identify if a cable is a multimode or single-mode fiber cable, but the color is not a reliable method. The TIA-598C standard suggests the outer coating to be yellow for single mode fiber and orange for multimode fiber for civilian applications. This guideline is not always implemented as illustrated in Figure 1-11, which shows a blue cable. The reliable method is to look at the specifications of the cable printed on the outer coating of the cabling. See also Figure 1-12 on page 10 and Figure 1-13 on page 10.

Figure 1-11 Blue 62.5 micron MMF cable

Figure 1-12 Yellow SMF cable

Figure 1-13 Orange 50 micron MMF cable

Connector types

The most common connector type for fiber optic media used in networking today is the LC connector, which is shown in Figure 1-14.

Figure 1-14 LC fiber connector

Other connectors that are commonly encountered in Ethernet networks are the SC connector (Figure 1-15), and the ST connector (Figure 1-16).

Figure 1-15 SC fiber connector

Figure 1-16 ST fiber connectors

Transceivers

A *transceiver* or *transmitter/receiver* is the fiber optic port of a device. It is where the fiber optic cables connect. Occasionally a device might have an integrated transceiver, which limits the flexibility in the type of cabling that can be used. Most devices provide a slot for a modular transceiver to be inserted, providing flexibility for single or multimode implementations to be selected.

Some equipment might use a larger transceiver known as a *Gigabit Interface Converter* (GBIC), which is shown in Figure 1-17. As technology advancements have been made, smaller transceivers have been introduced providing much higher port density, such as small form-factor pluggable (SFP), 10-Gigabit SFP+, or 10-Gigabit SFP-XFP. See also Figure 1-18 to compare the different transceivers.

Figure 1-17 Gigabit Interface Converter

Figure 1-18 From left to right: SFP-MMF, SFP-SMF, SFP+-MMF, XFP-MMF, and XFP-SMF

1.3.2 Components

The Layer 1 components in an Ethernet are devices that provide physical signal connectivity of the devices.

Hub

A network *hub* has the same functional properties of a segment using coax cabling to interconnect devices. It operates as a share media segment.

The difference is that the hub allows for much easier cabling. The cables plug into the hub, and the hub internally forms the share media segment. A network hub connects multiple devices as a single shared media segment. All frames are viewable by all the devices attached to the hub because it forms a shared media segment.

Repeater

A network *repeater* is similar in function to a network hub in that it forms a common shared media segment with all of the attached devices. The difference is a repeater receives the incoming signal and then “repeats” or regenerates the signal at new signal quality and strength as it is sent out the other ports of the repeater.

In contrast, a *hub* is a passive device that allows the original signal to pass through to the other devices that are connected to the segment. A repeater is often used in environments in which signal loss occurs in a segment because of media length or other environmental factors.

1.3.3 Network segments

A *network segment* can be thought of as the local Layer 1 network to a device and any devices on the same common Layer 1 network. A formal definition of a network segment is: a portion of a network separated by a networking device such as a router, switch, or bridge.

A network segment can be one of two types, a shared media or switched segment.

Shared media segment

Devices on a shared media segment must arbitrate for use of the media because multiple devices cannot transmit at the same time. The devices must “share” the media; hence the name “shared media segment.” Because all devices are connected to the same media (wire), all devices see communications from the other devices (see Figure 1-19 on page 13). The more devices that are attached to the shared media segment means that more devices are arbitrating for use of the segment to transmit. Use of this type of segment provides for a network that scales well.

Figure 1-19 Shared media segment

The Carrier Sense Multiple Access/Collision Detection (CSMA/CD) access method is the protocol that Ethernet devices follow when operating in half duplex mode which is required for a shared media segment. The CSMA/CD Access Method has three parts.

Carrier Sense requires a device to “listen” on the media and wait until no other transmissions are occurring before attempting to transmit. With multiple access, multiple nodes can be connected to the same media at the same time.

Collision Detection recognizes when two nodes have attempted to transmit at the same time and notifies the other nodes on the segment that the previous transmission is invalid. This notification function is known as *jamming*. Jamming consists of sending a special 32-bit sequence.

Switched segment

Devices on a switched segment can communicate directly to the network switch without any interference by other intermediate devices. Switched segments allow for full duplex communications.

1.3.4 Physical configuration parameters

When we discuss the physical layer (Layer 1) properties, we consider elements such as line speed and duplex.

Speed

Speed in an Ethernet refers to rates such as 10 Mbps, 100 Mbps, 1000 Mbps (1 Gbps), or 10 Gbps. The standards for higher speeds such as 40 and 100 Gbps are in draft form now.

Duplex

Duplex modes are either full or half duplex. *Half duplex* is when a device can only send or receive at a given time (Figure 1-20), while *full duplex* devices can send and receive at the same time (Figure 1-21). Also, when devices are connected to a shared media segment all devices must operate in half-duplex mode.

Figure 1-20 Half-duplex mode

Figure 1-21 Full-duplex mode

Autonegotiation

In an Ethernet, the speed and duplex of a device attached to a segment must match. Autonegotiation of the speed and duplex of a device usually works well, but it is not 100% reliable. The problems usually occur with older 10/100 devices. Newer devices rarely have an issue negotiating with each other.

One step to reduce negotiation problems is to ensure that both devices on a switch segment are configured the same. Either configure both devices for autonegotiation, or “hard code” (manually configure) both the speed and duplex settings of both devices to the same settings.

1.3.5 Power over Ethernet

Power over Ethernet (PoE) is the implementation of IEEE 802.3af, which allows both data and electric power to pass over a copper Ethernet LAN cable. With this technology, Voice over Internet Protocol (VoIP) telephones, wireless access points, video cameras, and point-of-sale devices can safely receive power from the same access ports that are used to connect personal computers to the network.

Power management mode

You can use the power management mode to determine the number of interfaces that can be provided with power. The following factors constitute the power management mode:

Per port limit (PPL) The factor that decides the maximum power consumption permitted on a particular interface. If the power consumption by the powered device exceeds the specified value, PoE is shut down over that interface.

Power allocated for each interface
The factor that ensures that a certain amount of power is reserved for an individual interface from the total power budget for all interfaces. At

any point, if the total of the allocated power for all interfaces exceeds the total budget, the lower priority interfaces are turned off and the power allocated for those interfaces drops to zero.

Power management has two modes:

Static The power allocated for each interface can be configured. The PPL value is the maximum value configured per interface. This mode is the default power management mode.

Class The power allocation for interfaces is determined based on the class of powered device connected. PPL is the maximum power value of the class of the powered device connected to the interface. The power allocated per interface is the maximum power of the powered device class, except for classes 0 and 3. For class 0 and class 3 powered devices, the momentary power consumption is considered as the power allocated for that interface. Therefore, PPL and power allocated per interface values change based on the powered device connected to the interface.

Classes of powered devices

A powered device is classified based on the maximum power that it draws across all input voltages and operational modes. The most common class is 0, in which the switch allows a maximum draw of 15.4 W per port. The switch provides 15.4 W at the port to guarantee enough power to run a device, after accounting for line loss. Consider the following example:

```
15.4 W – power loss (16%) = 12.95 W
```

All 802.3af-compliant powered devices require no more than 12.95 watts.

Table 1-2 lists the classes of powered devices and associated power levels.

Table 1-2 PoE classes

Class	Usage	Minimum power levels output from PoE port	Range of maximum power required by the powered device
0	Default	15.4 W	0.44 through 12.95 W
1	Optional	4.0 W	0.44 through 3.84 W
2	Optional	7.0 W	3.84 through 6.49 W
3	Optional	15.4 W	6.49 through 12.95 W

1.4 Layer 2 networking concepts and terminology

OSI Layer 2 or the Data Link Layer provides the functional means for data transfer between adjacent nodes in the network. Layer 2 also provides the lowest level of addressability in an Ethernet network using MAC addresses. This section provides information about the fundamental concepts and terminology at Layer 2.

1.4.1 Frame structure

A *frame* is the unit of transmission at the Layer 2 operations of a network. A frame is a series of bits. The groups of these bits are known as *fields* in the frame and have specific purposes (Figure 1-22).

Figure 1-22 Ethernet frame

A frame is composed of the following fields

Preamble (7 bytes) An alternating sequence of 1 and 0 bit values that are used to synchronize the devices.

Start Frame Delimiter (SFD, 1 byte)
A byte containing 10101011, which signals the start of the frame that follows.

Destination Address (6 bytes)
A MAC address of the destination device for the frame. A destination address has three forms:

- Unicast: Unique address for an individual device
- Multicast: Address for a group of devices
- Broadcast: Special multicast address destined for all devices

Source Address (6 bytes)
MAC address of the source device of the frame.

Ethernet Protocol Type or Length (2 bytes)
Use of this field is the distinction between 802.3 and Ethernet II (DIX).

- Values of x05DC (1500 decimal) or less indicate an 802.3 frame type and the value is the length of Payload field.
- Values of x0600 and higher indicate an Ethernet II (DIX) frame type and the value is the Protocol Type.

Payload/Data (46 - 1500 bytes)
The data to be transferred from source to destination device. It is normally up to 1500 bytes. If the data is less than 46 bytes, then “padding” must be added to maintain a minimum frame size.

Frame Check Sequence (FCS, 4 bytes)
Four-byte cyclical redundancy check (CRC) value for error checking. The value is calculated from the DA field through the Payload field by the sending device. The receiving device makes the same calculation and compares the value with the value in the FCS field.

The maximum payload size for a standard Ethernet frame is 1500 bytes. The total frame size of 1518 bytes.

Notice that this frame structure does not include any field to identify virtual local area network (VLAN) membership or frame priority. Such a frame might be known as an “untagged” frame. For information about tagged frames, see “Tagged frames” on page 24.

1.4.2 Jumbo frames

A *jumbo frame* is an expansion of the frame size up to approximately 9K. The maximum size support is vendor-dependent. Although some vendors support beyond 9K, 9K is a common maximum size.

With jumbo frames, devices provide better performance because they do not need to process as many frames (frame headers). The result is less processor utilization and less bandwidth to transmit header information and less idle time of link because of interframe gaps.

Concurrent use of jumbo and standard frames sizes in some networks can impact performance of latency-sensitive applications. End-to-End network devices must support jumbo frames. Otherwise frames are discarded.

1.4.3 Interframe gap

Ethernet devices must allow an idle period between frame transmissions (Figure 1-23). The minimum idle time is 96 bit times and is the amount of time required to send 96 bits.

Figure 1-23 Interframe cap

The interframe gap (IFG) is also referred to as *interframe spacing* or *interpacket gap* (IPG).

1.4.4 MAC addresses

Each frame contains a source and a destination MAC address. The device sending the frame includes its MAC address as the source MAC address, and the destination MAC address is the MAC address of the device the frame is targeted. The NIC of each device communicating in the network has a MAC address that was assigned by the manufacturer.

Figure 1-24 shows the various ways in which a manufacturer might show a MAC address.

00:13:10:17:FB:F8 00-13-10-17-FB-F8 0013.1017.FBF8

- The first three octets (highlighted in red) are assigned by IEEE to NIC manufacturer as the OUI (Organizationally Unique Identifier).
- Last three octets (highlighted in green) are assigned by a manufacturer while maintaining the uniqueness of the address.

Figure 1-24 MAC addresses

To search for the registered owner of an Organizationally Unique Identifier (OUI), see the IEEE Standards Association website at:

http://standards.ieee.org/regauth/oui/index.shtml

Within the first byte of a MAC address, 2 bits provide additional information as illustrated in Figure 1-25.

00:13:10:17:FB:F8 or 00-13-10-17-FB-F8
or 0013.1017.FBF8

The first octet in more detail:

b7	b6	b5	b4	b3	b2	b1	b0

- B0 – multicast bit:
  - 0 – unicast
  - 1 – multicast
- B1 – locally administered address bit:
  - 0 – globally unique (burned-in address)
  - 1 – locally administered

Figure 1-25 Details of a MAC address

Looking at the first byte of a MAC address in more detail, bit 0 and bit 1 have specific functions. Bit 0 of the first octet is the multicast bit. If the value of the bit is 0, the frame is a unicast frame. If the value is 1, the frame is a multicast frame.

Bit 1 of the first octet is the locally administered address bit. If the value of the bit is 0, the MAC address is a globally unique address (manufacturer's assigned or burned-in address). If the value is 1, the MAC address is a locally administered MAC address that is assigned by the user or administrator.

Types of Layer 2 traffic

An Ethernet has three types of Layer 2 traffic. The type of traffic is determined by the destination MAC address of the frame:

- Unicast (one to one)

  Unicast is traffic with a specific device's MAC address as the destination MAC address in the frame. The address can be either the burned-in or a locally administered MAC address. The device with the same MAC address as the destination processes the incoming frame.

- Multicast (one to many)

  Multicast is traffic with a multicast MAC address as the destination MAC in the frame. The multicast MAC address is sometimes referred to as a *group MAC address*. The actual destination devices are any devices configured to process frames with that specific group MAC address. Consider the following examples:

  – `01-80-C2-00-00-00` used by the Spanning Tree Protocol (IEEE 802.1d)
  – `01-00-5E-xx-xx-xx` used by IPv4 Multicast Addresses

- Broadcast (one to every)

  Broadcast is traffic with a special case of multicast frame in which the group address is a special address, `FF-FF-FF-FF-FF-FF`, as the destination MAC in the frame. All Layer 2 devices in the network must process the broadcast frame.

1.4.5 Bridging or Layer 2 switching

A bridge connects multiple segments at the Data Link Layer (OSI Layer 2) as shown in Figure 1-26.

Figure 1-26 Bridge or Layer 2 switch

Another way to express the same idea is a bridge that maintains one broadcast domain (Layer 2) while isolating collision domains (Layer 1). Bridges began as PCs with multiple NICs installed with a special bridging application running that performs the bridging function. MAC addresses of attached devices are learned on each port.

What is the difference between a bridge and a Layer 2 switch? From a functional standpoint, the answer is *that there is no difference*. A simple way to explain the difference is that a bridge is a device that performs the function of maintaining the broadcast domain while isolating the collision domain.

A Layer 2 switch is a device that implements the bridging function primarily in hardware. Layer 2 switches have most of the bridging function incorporated into the chip sets that forward the frames. This method typically provides for wire speed forwarding rates of traffic. The speed and port density of a switch is much more that the old bridge implementations on PCs.

As stated previously, the MAC addresses of devices attached to a switch are added to the MAC table of the switch along with the port on which the MAC address was learned. This method is accomplished by monitoring incoming source MAC addresses and adding any new MAC addresses to the table. If a device moves from one port to another, the MAC entry in the table is updated with the new port. Also, if a device is powered off, removed from the network, or is not sending traffic in the network, the MAC entry is removed. The removal occurs after the time at which any traffic received from this MAC address has exceeded an expiration time value of the switch.

Frame forwarding

This section examines how a Layer 2 switch forwards traffic in different cases.

Multicast or broadcast frames

The first case, which is illustrated in Figure 1-27, describes the process for forwarding a multicast or broadcast frame sent from device with MAC A attached to port 1 of the switch. Upon receiving the frame, the switch copies the frame and sends it to all other ports.

Figure 1-27 Broadcast/multicast frame forwarding

Unicast frame: MAC learned

The next case, which is illustrated in Figure 1-28, describes the process for forwarding a unicast frame with the destination MAC G received from the device on port 1 of the switch. Upon receiving the frame, the switch queries the MAC address table in the switch for the destination MAC address. If the MAC address is found in the table, the frame is forwarded out the port associated with the MAC address in the table. In this example, the frame is forwarded to port 4 because the MAC table has an entry for MAC G on port 4.

The MAC address entries are dynamically populated by monitoring the source MAC address of frames coming into the ports. MAC entries age out of the table so that devices that are not sending frames will be removed. The age-out time is vendor-dependent; some vendors allow for the time-out value to be a configurable parameter. The age-out time can also vary depending on other protocols or features that are enabled on the switch.

If a device is moved from one port to another port, the MAC address is learned, the old entry in the MAC table is deleted, and a new entry is added with the new port.

Figure 1-28 Unicast frame forwarding, MAC learned

Unicast frame: MAC unknown

The last case, which is illustrated in Figure 1-29, describes the process for forwarding a unicast frame sent from a device with a destination of MAC C (not in the MAC address table of the switch) sent from a device with MAC A on port 1 of the switch. Upon receiving the frame, the switch queries the MAC address table in the switch for the destination MAC address. If the MAC is *not* found in the table, the frame is copied and forwarded to all other ports. In this example, the frame is forwarded to all other ports of the switch.

This method is not the same as a broadcast because the frames that are forwarded are unicast frames (frame with the specific MAC address of the destination device). Some vendors refer to the process as *flooding*. The frames are sent down all paths to reach the destination device.

Figure 1-29 Unicast frame forwarding, MAC unknown

Internet Group Management Protocol

Internet Group Management Protocol (IGMP) is a protocol that provides management of multicast traffic in a Layer 2 network. With this protocol, multicast traffic for a specific multicast group can only be sent to ports of switches that have clients that use the group's multicast traffic.

1.4.6 Virtual local area network

A VLAN is a networking concept in which a network is logically divided into smaller virtual LANs. The Layer 2 traffic in one VLAN is logically isolated from other VLANs as illustrated in Figure 1-30.

Figure 1-30 Isolation at Layer 2

Figure 1-31 shows two methods for maintaining isolation of VLAN traffic between switches.

Figure 1-31 VLAN tagging

The first method uses a single link for each VLAN. This method does not scale well because it uses many ports in networks with multiple VLANs and multiple switches. Also, this method does not use link capacity efficiently when traffic in the VLANs is not uniform.

The second method is VLAN tagging over a single link in which each frame in tagged with its VLAN ID. This method is highly scalable because only a single link is required to provide connectivity to many VLANs, which provides for better utilization of the link capacity when VLAN traffic is not uniform.

The protocol for VLAN tagging of frames in a LAN environment is defined by the IEEE 802.1 P/Q standard.

Inter-Switch Link (ISL): ISL is another protocol for providing the VLAN tagging function in a network. This protocol is not compatible with the IEEE 802.1P/Q standard.

Tagged frames

The IEEE 802.1P/Q standard provides a methodology for added information such as VLAN membership and priority to the frame as shown in Figure 1-32.

Figure 1-32 IEEE 802.1 P/Q tagged Ethernet frame

The standard provides an additional 4 bytes of information to be added to each Ethernet frame. A frame including this extra information is known as a *tagged frame*.

The 4 byte tag has four component fields:

- A type field that is 2-bytes long with the hexadecimal value of x8100 to identify the frame as an 802.1P/Q tagged frame
- A priority field of 3-bits long to allow a priority value of eight different values to be included in the tag; has the “P” portion of the 802.1P/Q standard
- A Canonical Format Indicator field that is 1-bit long to identify when the contents of the payload field are in canonical format
- A VLAN ID field that is 12-bits long to identify which VLAN the frame is a member of, with 4096 different VLANs possible

1.4.7 Interface VLAN operation modes

Interfaces on a switch can operate in two VLAN modes, single VLAN mode or multiple VLAN mode.

Single VLAN mode

Single VLAN mode operation is also referred to as *access mode*. A port operating in this mode is associated with a single VLAN. Incoming traffic does not have any VLAN identification. While the untagged frames enter the port, the VLAN identification for the VLAN configured for the port is added to the inbound frames.

Switch ports: Some vendors use terms other than access mode for ports operating in single VLAN mode. The switch ports of those vendors might be configured to operate in single VLAN mode by configuring a Port VLAN ID (PVID) and adding the port as a member of the VLAN.

Multiple VLAN mode

Multiple VLAN mode operation is also referred to as *trunk mode*. A port operating in this mode can receive frames that have VLAN tags. The port is also configured with VLANs to which the port is allowed to send and receive frames.

With the IEEE 802.1Q specification, untagged traffic on a multi-VLAN port can be associated with a single VLAN, which is referred to as the “native” VLAN for the port (Figure 1-33). By using this provision, traffic with no VLAN tag can be received and associated with the VLAN that is configured as the PVID or native VLAN. Outbound traffic for this VLAN on a port configured in this manner is transmitted with no tag so that the receiving device can receive the frame in an untagged format.

This method provides compatibility with existing devices or devices that are configured in single VLAN mode and attached to a port configured as a multi-VLAN port.

Figure 1-33 Multiple VLAN mode link

Variations in the meaning of trunk: The term *trunk* is used to express different ideas in the networking industry. When using this term, keep in mind that others might use the term in a different manner. The term trunk can mean a port operating in multiple VLAN mode or it can mean a link aggregated port.

1.4.8 Link aggregation

Link aggregation combines multiple physical links to operate as a single larger logical link. The member links no longer function as independent physical connections but as members of the larger logical link (Figure 1-34).

Figure 1-34 Link aggregation

Link aggregation provides greater bandwidth between the devices at each end of the aggregated link. Another advantage of link aggregation is increased availability, because the aggregated link is composed of multiple member links. If one member link fails, the aggregated link continues to carry traffic over the remaining member links.

Each of devices interconnected by the aggregated link uses a hashing algorithm to determine on which of the member links frames will be transmitted. The hashing algorithm might use varying information in the frame to make the decision. This algorithm might include a source MAC, destination MAC, source IP, destination IP and more. It might also include a combination of these values.

1.4.9 Spanning Tree Protocol

Spanning Tree Protocol (STP) provides Layer 2 loop prevention and is commonly in different forms, such as existing STP, Rapid STP (RSTP), Multiple STP (MSTP), and VLAN STP (VSTP). RSTP is a common default STP. It provides faster convergence times than STP. However, some existing networks require the slower convergence times of basic STP.

The operation of STP

STP uses Bridge Protocol Data Unit (BPDU) packets to exchange information with other switches. BPDUs send out hello packets at regular intervals to exchange information across bridges and detect loops in a network topology.

Two types of BPDUs are available:

- Configuration BPDUs

  These BPDUs contain configuration information about the transmitting switch and its ports, including switch and port MAC addresses, switch priority, port priority, and port cost.

- Topology Change Notification (TCN) BPDUs

  When a bridge must signal a topology change, it starts to send TCNs on its root port. The designated bridge receives the TCN, acknowledges it, and generates another one for its own root port. The process continues until the TCN reaches the root bridge.

STP uses the information provided by the BPDUs to elect a root bridge, identify root ports for each switch, identify designated ports for each physical LAN segment, and prune specific redundant links to create a loop-free tree topology. All leaf devices calculate the best path to the root device and place their ports in blocking or forwarding states based on the best path to the root. The resulting tree topology provides a single active Layer 2 data path between any two end stations.

Rapid Spanning Tree Protocol

RSTP provides better re-convergence time than the original STP. RSTP identifies certain links as point to point. When a point-to-point link fails, the alternate link can make the transition to the forwarding state.

An RSTP domain has the following components:

Root port	The “best path” to the root device.
Designated port	Indicates that the switch is the designated bridge for the other switch connecting to this port.
Alternate port	Provides an alternate root port.
Backup port	Provides an alternate designated port.

RSTP was originally defined in the IEEE 802.1w draft specification and later incorporated into the IEEE 802.1D-2004 specification.

Multiple Spanning Tree Protocol

Although RSTP provides faster convergence time than STP, it still does not solve a problem inherent in STP, that all VLANs within a LAN must share the same spanning tree. To solve this problem, we use MSTP to create a loop-free topology in networks with multiple spanning-tree regions.

In an MSTP region, a group of bridges can be modeled as a single bridge. An MSTP region contains multiple spanning tree instances (MSTIs). MSTIs provide different paths for different VLANs. This functionality facilitates better load sharing across redundant links.

An MSTP region can support up to 64 MSTIs, and each instance can support anywhere from 1 through 4094 VLANs.

MSTP was originally defined in the IEEE 802.1s draft specification and later incorporated into the IEEE 802.1Q-2003 specification.

VLAN Spanning Tree Protocol

With VSTP, switches can run one or more STP or RSTP instances for each VLAN on which VSTP is enabled. For networks with multiple VLANs, VSTP enables more intelligent tree spanning. This level of tree spanning is possible because each VLAN can have interfaces enabled or disabled depending on the paths that are available to that specific VLAN.

By default, VSTP runs RSTP, but you cannot have both stand-alone RSTP and VSTP running simultaneously on a switch. VSTP can be enabled for up to 253 VLANs.

BPDU protection

BPDU protection can help prevent STP misconfigurations that can lead to network outages. Receipt of BPDUs on certain interfaces in an STP, RSTP, VSTP, or MSTP topology, can lead to network outages.

BPDU protection is enabled on switch interfaces connected to user devices or on interfaces on which no BPDUs are expected, such as edge ports. If BPDUs are received on a protected interface, the interface is disabled and stops forwarding frames.

Loop protection

Loop protection increases the efficiency of STP, RSTP, VSTP, and MSTP by preventing ports from moving into a forwarding state that might result in a loop opening in the network.

A blocking interface can transition to forwarding state in error if the interface stops receiving BPDUs from its designated port on the segment. Such a transition error can occur when there is a hardware error on the switch or software configuration error between the switch and its neighbor.

When loop protection is enabled, the spanning tree topology detects root ports and blocked ports and ensures that both keep receiving BPDUs. If a loop-protection-enabled interface stops receiving BPDUs from its designated port, it reacts as it might react to a problem with the physical connection on this interface. It does not transition the interface to a forwarding state, but instead transitions it to a loop-inconsistent state. The interface recovers and then transitions back to the spanning-tree blocking state as soon as it receives a BPDU.

You must enable loop protection on all switch interfaces that have a chance of becoming root or designated ports. Loop protection is most effective when enabled in the entire switched network. When you enable loop protection, you must configure at least one action (**`alarm`**, **`block`**, or both).

An interface can be configured for either loop protection or root protection, but not for both.

Root protection

Root protection increases the stability and security of STP, RSTP, VSTP, and MSTP by limiting the ports that can be elected as root ports. A root port elected through the regular process has the possibility of being wrongly elected. A user bridge application running on a PC can also generate BPDUs and interfere with root port election. With root protection, network administrators can manually enforce the root bridge placement in the network.

Root protection is enabled on interfaces that should not receive superior BPDUs from the root bridge and should not be elected as the root port. These interfaces become designated ports and are typically on an administrative boundary. If the bridge receives superior STP BPDUs on a port that has root protection enabled, that port transitions to a root-prevented STP state (inconsistency state), and the interface is blocked. This blocking prevents a bridge that should not be the root bridge from being elected the root bridge. After the bridge stops receiving superior STP BPDUs on the interface with root protection, the interface returns to a listening state, followed by a learning state, and ultimately back to a forwarding state. Recovery back to the forwarding state is automatic.

When root protection is enabled on an interface, it is enabled for all the STP instances on that interface. The interface is blocked only for instances for which it receives superior BPDUs.

Otherwise, it participates in the spanning tree topology. An interface can be configured for either root protection or loop protection, but not for both.

1.4.10 Link Layer Discovery Protocol

Link Layer Discovery Protocol (LLDP) is a vendor independent protocol for network devices to advertise information about their identity and capabilities. It is referred to as *Station and Media Access Control Connectivity Discovery*, which is specified in the 802.1ab standard. With LLDP and Link Layer Discovery Protocol–Media Endpoint Discovery (LLDP-MED), network devices can learn and distribute device information on network links. With this information, the switch can quickly identify various devices, resulting in a LAN that interoperates smoothly and efficiently.

LLDP-capable devices transmit information in Type Length Value (TLV) messages to neighbor devices. Device information can include specifics, such as chassis and port identification and system name and system capabilities. The TLVs use this information from parameters that have already been configured in Junos software from Juniper Networks.

LLDP-MED goes one step further, exchanging IP-telephony messages between the switch and the IP telephone. These TLV messages provide detailed information about the PoE policy. With the PoE Management TLVs, the switch ports can advertise the power level and power priority needed. For example, the switch can compare the power needed by an IP telephone running on a PoE interface with available resources. If the switch cannot meet the resources required by the IP telephone, the switch can negotiate with the telephone until a compromise on power is reached.

The switch also uses these protocols to ensure that voice traffic gets tagged and prioritized with the correct values at the source itself. For example, 802.1p class-of-service (COS) and 802.1Q tag information can be sent to the IP telephone.

1.4.11 LLDP TLVs

Basic TLVs include the following information:

Chassis identifier The MAC address associated with the local system.

Port identifier The port identification for the specified port in the local system.

Port description The user-configured port description. The port description can be a maximum of 256 characters.

System name The user-configured name of the local system. The system name can be a maximum of 256 characters.

System description The system description containing information about the software and current image running on the system. This information is not configurable, but taken from the software.

System capabilities The primary function performed by the system. The capabilities that system supports, for example, bridge or router. This information is not configurable, but is based on the model of the product.

Management address The IP management address of the local system.

Additional 802.3 TLVs include the following details:

Power via MDI A TLV that advertises MDI power support, Power Sourcing Equipment (PSE) power pair, and power class information.

MAC/PHY configuration status
A TLV that advertises information about the physical interface, such as autonegotiation status and support and MAU type. The information is not configurable, but based on the physical interface structure.

Link aggregation A TLV that advertises if the port is aggregated and its aggregated port ID.

Maximum frame size A TLV that advertises the maximum transmission unit (MTU) of the interface sending LLDP frames.

Port VLAN A TLV that advertises the VLAN name configured on the interface.

LLDP-MED provides the following TLVs:

LLDP MED capabilities
A TLV that advertises the primary function of the port. The capabilities values range 0 through 15:

0	Capabilities
1	Network policy
2	Location identification
3	Extended power via MDI-PSE
4	Inventory
5 through 15	Reserved

LLDP-MED device class values

0	Class not defined
1	Class 1 device
2	Class 2 device
3	Class 3 device
4	Network connectivity device
5 through 255	Reserved

Network policy A TLV that advertises the port VLAN configuration and associated Layer 2 and Layer 3 attributes. Attributes include the policy identifier, application types, such as voice or streaming video, 802.1Q VLAN tagging, and 802.1p priority bits and Diffserv code points.

Endpoint location A TLV that advertises the physical location of the endpoint.

Extended power via MDI
A TLV that advertises the power type, power source, power priority, and power value of the port. It is the responsibility of the PSE device (network connectivity device) to advertise the power priority on a port.

1.5 Layer 3 networking concepts and terminology

At the Layer 2 level of the network, the function provided a means of communication to adjacent nodes in the network or node-to-node communications. Layer 3 functions of the Ethernet network provide the means for communications from the source to destination. This includes path determination or routing between Layer 3 logical networks. For Internet Protocol networks, the logical Layer 3 network are also known as *subnets*. A common network design is to have a single IP subnet configured for each VLAN as illustrated in Figure 1-35.

Figure 1-35 Single IP subnet per VLAN

Another design option is to have multiple IP subnets configured on a single VLAN as illustrated in Figure 1-36.

Figure 1-36 Multiple IP subnets per VLAN

1.5.1 IP routing

A router forwards Layer 3 packets from one logical network (IP subnet) to another as shown in Figure 1-37. A router provides for connectivity at the Layer 3 level of the network while broadcasts are isolated to the separated Layer 2 networks or broadcast domains.

Figure 1-37 Layer 3 Routing

The IP routing or IP forwarding process can have several steps depending on the features and security functions that are enabled. Figure 1-38 shows the core process that is followed. This diagram does not include every step but shows a high-level view of the forwarding process.

Figure 1-38 IP routing or IP forwarding process

1.5.2 Address Resolution Protocol

Address Resolution Protocol (ARP) provides the IP with the MAC addresses of adjacent nodes in the network as communications require. IP devices build and maintain a dynamic IP to MAC address mapping in the memory known as an *ARP table* or *ARP cache*.

The MAC address of a target host is learned by the transmitting host sending a request as a Layer 2 broadcast to all devices. This request is known as an *ARP request*. The ARP request contains the IP and MAC address of the requesting device for the response to be sent.

The device with the target IP address responds to the request with the target's MAC address. This response is known as an *ARP reply*. The sending station receives the ARP reply and updates the ARP cache. Now the sending device has the information necessary to send the packet to the target device.

1.5.3 IPv4 addressing

IPv4 addresses are 32 bits in length and are used to logically address devices at the Layer 3 level in a network. The 32 bits are represented in a “dotted quad” notation in which the decimal value of groups of 8 bits of the 32 bits address are separated by decimals as illustrated in Figure 1-39.

Figure 1-39 IPv4 addresses

IP address classes

IPv4 address were initially assigned to users on a class basis. Different classes provide a different number of usable addresses as shown in Table 1-3.

Table 1-3 IPv4 address classes

IP class	Address range	Subnet mask	Number of host addresses
A	1.x.x.x - 127.x.x.x	255.0.0.0	16,777,214
B	128.0.x.x - 191.255.x.x	255.255.0.0	65534
C	192.0.0.x - 223.255.255.x	255.255.255.0	254
D	224.0.0.0 - 239.255.255.255	Multicast	
E	240.0.0.0 - 255.255.255.255	Reserved	

IPv4 address consumption: In the 1980s, fear of consuming all of the IPv4 addresses available in the Internet fueled ideas about to how to overcome the limitation of only having approximately 4 000 000 000 000 IPv4 address available for use. Some solutions were implemented to provide methods to more efficiently use the IPv4 address space.

Classless Inter-Domain Routing

Classless Inter-Domain Routing (CIDR) was introduced in 1993 as a replacement to the classful IP address syntax. CIDR provides greater flexibility in dividing IP address ranges into separate networks and allows for a more efficient use of the IPv4 addresses. IP address blocks are denoted by the IP address and the number of bits in the network mask (10.69.71.91/24).

Subnet mask

An IP subnet mask is a 32-bit mask that is used in IPv4 to identify the network portion of the IP address. The remaining bits are the host portion of the address as illustrated in Figure 1-40.

- A subnet mask identifies the bits of the 32 bit IP address that are the network portion of the address and which are the host portion of the address.

  Address:
  11000000101010000100011001111101
  Mask:
  11111111111111111111111100000000
  ^----------Network----------^^--Host--^

- Written in different formats:
  - 192.168.70.125 255.255.255.0
  - 192.168.70.125 FF FF FF 00
  - 192.168.70.125/24
- For each subnet, two address are reserved (cannot be assigned to devices).
  - All host bits that are zeroes indicate the network ID for the subnet.
  - All host bits that are ones indicate the broadcast address for the subnet.

Figure 1-40 Subnet mask

When an IP device communicates with another IP host, the device compares the bits of the local IP address masked by the subnet masked with the bits of the target IP address masked by the same subnet mask. If the masked bits of the two addresses match, the target address is on the same subnet as the sending device. If they do not match, the target device is on a remote subnet.

In each subnet, two addresses are reserved that cannot be assigned as host addresses:

- Network ID

  The network ID for a subnet is the address in which hosts bits of the masked address are all *zeros*. The network ID is the first address in the subnet. The network ID is used to reference the subnet in routing tables.

- Broadcast Address

  The broadcast address for a subnet is the address in which hosts bits of the masked address are all *ones*. The broadcast address is the last address in the subnet.

1.5.4 IPv6 addressing

IPv6 was adopted in 1994 by the Internet Engineering Task Force (IEFT) as IPng or IP Next Generation and later became known as IPv6. IPv6 has many differences from IPv4. One of the most significant and easily recognizable differences is the IP address as illustrated in Figure 1-41.

- 128-bit address, written as 8 groups of 4 hexadecimal digits with each group separated by colons.

```
2001:0db8:0000:0000:0000:0000:c0a8:467d
^----Network Prefix----^^------------Host---------^
```

- 64-bit network prefix portion and 64-bit host portion
- Acceptable notations
  - 2001:0db8:0000:0000:0000:0000:c0a8:467d
  - 2001:0db8:0:0:0:0:c0a8:467d
  - 2001:0db8::0000:0000:0000:c0a8:467d
  - 2001:db8::c0a8:467d

Figure 1-41 IPv6 addresses

For details about IPv6 addressing, see Internet Engineering Task Force (IETF) RFC 4291 at the following address:

http://tools.ietf.org/html/rfc4291

Route table

A *route table* is a list of the known subnets in a network and the desired path to reach each subnet. The tables contain subnets learned from local interfaces, static routes, or learned routes from routing protocols. For each subnet, the route table contains the IP address of the next router in the path if the subnet is a remote subnet or the interface if the subnet is a local subnet to the router.

Routing protocols

Routing protocols provide a dynamic method for updating route tables in the routers of an IP network. If routing protocols are not used, the routes must be manually configured by the use of static routes. Many routing protocols are used in the industry.

Common routing protocols include the following types:

Routing Information Protocol (RIP)
A distance-vector routing protocol. RIP uses hop count for path selection. The path with the fewest number of routers is the preferred path.

Open Shortest Path First (OSPF)
A link-state protocol. OSPF uses path cost for path selections. A designated router runs Dijkstra's algorithm to calculate the shortest path tree for each OSPF area.

Border Gateway Protocol (BGP)
A path-vector protocol. BGP uses a path, rule sets, and network policies for path selection. It supports CIDR and uses route aggregation to decrease the size of routing tables.

Default gateway

A *default gateway* is the device on a subnet that passes traffic from the local subnet to devices on a remote subnet. When the destination IP address of a packet is determined to be on a remote subnet, the packet is sent to the default gateway for routing of the packet along the path to the destination device.

Default route

A *default route* is a parameter that is configured in a router that indicates where to forward packets with destination subnets that are not contained in the route table of the router.

2

Introduction to IBM j-type Ethernet switches and routers

This chapter provides information about the IBM j-type Ethernet switches and routers portfolio. This portfolio is added to the IBM product line through an original equipment manufacturer (OEM) agreement with Juniper Networks.

This introductory chapter includes the following topics:

- IBM j-type e-series Ethernet switches
- IBM j-type m-series Ethernet routers
- Junos operating system
- Simplification of the IBM Data Center Network solution

© Copyright IBM Corp. 2011. All rights reserved.

2.1 IBM j-type e-series Ethernet switches

The IBM j-type e-series Ethernet switch portfolio is made up of the following models:

- IBM Ethernet Switch J48E
- IBM Ethernet Switch J08E
- IBM Ethernet Switch J16E

The first system is a fixed port unit as shown in Figure 2-1.

Figure 2-1 IBM Ethernet Switch J48E

The other two IBM j-type e-series Ethernet switches (shown in Figure 2-2) are chassis based.

Figure 2-2 IBM Ethernet Switches J08E and J16E

If you are familiar with the Juniper Networks products, Table 2-1 provides a cross-reference between the IBM j-type e-series Ethernet Switches and the models offered by Juniper Networks.

Table 2-1 Cross-reference of IBM j-type e-series Ethernet switches to Juniper Networks models

IBM description	IBM machine type and model	Juniper Networks model
IBM Ethernet Switch J48E	4273-E48	EX4200
IBM Ethernet Switch J08E	4274-E08	EX8208
IBM Ethernet Switch J16E	4274-E16	EX8216

The following sections provide in-depth information about each model.

2.1.1 IBM Ethernet Switch J48E

The IBM Ethernet Switch J48E is a high-performance, scalable Ethernet switch for top-of-rack and end-of-row deployments in a data center. It delivers high availability, high performance, and a single point of management in a compact, power-efficient one rack-unit (1RU) form factor system.

The IBM Ethernet Switch J48E combines the high availability and carrier-class reliability of modular systems with the economics and flexibility of stackable platforms. It provides a high-performance, scalable solution for top-of-rack and end-of-row applications in a data center.

By offering a full suite of Layer 2 and Layer 3 switching capabilities as part of the base software, the IBM Ethernet Switch J48E can also satisfy a variety of high-performance branch and campus deployments. A single 48-port switch can be deployed initially. Then as requirements grow, integrated Virtual Chassis technology allows up to 10 switches to be interconnected over a 128 gigabit-per-second (Gbps) back plane and managed as a single device. This capability delivers a scalable, pay-as-you-grow solution for supporting escalating customer requirements. Optional Gigabit Ethernet (GbE) and 10 Gigabit Ethernet (10 GbE) uplink modules enable high-speed connectivity to aggregation- or core-layer switches.

The IBM Ethernet Switch J48E includes features for high availability, such as redundant, hot-swappable internal power supplies and field-replaceable, multi-blower fan trays to maximize uptime. In addition, the switch offers Class 3 Power over Ethernet (PoE). It delivers 15.4 watts on the first eight ports to support networked devices. Such devices include top-of-rack security cameras and wireless LAN (WLAN) access points for low-density converged networks.

Architecture and key components

The IBM Ethernet Switch J48E is a single rack-unit device that offers 48 10/100/1000BASE-T ports. The eight PoE ports include built-in Link Layer Discovery Protocol-Media Endpoint Discovery (LLDP-MED) services. These services provide a standards-based component discovery mechanism. They make the IBM Ethernet Switch J48E ideal for datacenter access deployments where a few security cameras, WLAN access points, or other devices require power. The IBM Ethernet Switch J48E also runs the same Junos operating system from Juniper Networks, that runs on the IBM Ethernet Switch J08E and J16E. Use of this operating system helps to reduce capital and operational expenses across the datacenter infrastructure.

Pay-as-you-grow scalability

With Virtual Chassis technology, a data center can interconnect as many as 10 IBM Ethernet Switch J48Es to satisfy connectivity requirements, delivering true chassis-like functionality in a compact, one rack-unit (RU) form factor. In a unique pay-as-you-grow model, a single 1RU top-of-rack switch can be deployed initially. Then as demands increase, up to nine more switches can be added incrementally for a total of 10 switches.

Resiliently interconnected through a 128 Gbps virtual backplane or front-panel 10 GbE uplink modules, a fully loaded Virtual Chassis configuration can support up to 480 10/100/1000BASE-T ports plus up to 20 10 GbE uplink ports. Virtual Chassis technology helps to lower capital expenses by requiring a lower up-front investment than existing chassis systems. It can also help dramatically reduce operating expenses by enabling up to 10 interconnected IBM Ethernet Switch J48Es to be operated and managed as a single logical device.

This incremental, pay-as-you-grow model, combined with the compact form factor of the IBM Ethernet Switch J48E, helps data centers to reduce up front and recurring rack space usage. It also saves on costly power and cooling.

Virtual Chassis

Virtual Chassis technology provides exceptional device and link high availability using the virtual backplane protocol and operating system software. Each set of interconnected switches automatically takes full advantage of multiple available Routing Engines to deliver graceful protocol restart. Graceful Routing Engine Switchover (GRES) and non-stop forwarding ensure uninterrupted operation in the rare event of an individual switch failure. For added device and link high availability, the IBM Ethernet Switch J48E can be configured to address many different requirements. For example, a single 10-switch Virtual Chassis configuration can be configured as two 5-switch Virtual Chassis configurations.

Location independence

Virtual Chassis technology can also be extended across the front-panel 10 GbE uplink ports to interconnect switches that are separated by more than a few meters. This configuration creates a single virtual switch that spans multiple telecommunications closets, floors, or even datacenter server racks.

A scalable IBM Ethernet Switch J48E top-of-rack deployment takes full advantage of Virtual Chassis technology. It uses a minimum amount of space with small form-factor switches that scale with high-density wire-speed ports, lowering heating and cooling costs while conserving space. With Virtual Chassis technology, up to 10 units can interoperate and be managed as a single device. In doing so, this technology dramatically simplifies configuration and management while reducing operational costs and simplifying cabling. Fewer uplinks are required when up to 10 top-of-rack switches are configured as a single virtual device, further lowering cost and complexity. Most importantly, the servers attached to the top-of-rack devices are interconnected by a single, high-bandwidth, low-latency switch. These servers do not need to rely on traffic going to an aggregation switch for server-to-server communications.

Configurations that require end-of-row deployments can also take advantage of Virtual Chassis technology. Such configuration can deploy a small form-factor IBM Ethernet Switch J48E that scales with high-density wire-speed ports as needed, lowering heating and cooling costs while conserving space.

Carrier-class reliability

The IBM Ethernet Switch J48E with Virtual Chassis technology provides the same high-availability features as modular chassis-based systems. Each switch supports internal redundant, load-sharing, hot-swappable power supplies, and a field-replaceable

hot-swappable fan tray with redundant blowers. Any of these items can fail without affecting operations.

Common features

The IBM Ethernet Switch J48E includes the following common features:

- Full Layer 2 and Layer 3 Ethernet switching
- Virtual Chassis technology

  With this technology, up to 10 switches can be interconnected as a single logical device, supporting up to 480 ports in top-of-rack or end-of-row applications for a data center.
- Interconnected switches in a Virtual Chassis configuration

  These switches share a single control plane and operating system and automatically assign master (active) and backup (hot-standby) Routing Engines
- Integrated application-specific integrated circuit (ASIC)-based Packet Forwarding Engine and integrated Routing Engine that deliver switching and routing functionality at wire rates
- Front-panel LCD display

  This panel offers a flexible interface for starting a device, rolling back configurations, reporting switch alarm and LED status, or restoring the switch to its default settings.
- Eight ports of IEEE 802.3af compatible Class 3 PoE
- Easy, centralized software upgrades
- The ability for all IBM j-type Ethernet switches and routers to run the modular, fault-tolerant Junos operating system

Specifications

Table 2-2 lists the specifications for the IBM Ethernet Switch J48E.

Table 2-2 IBM Ethernet Switch J48E specifications

Feature	Specification
Form factor	Fixed platform (single switch) A Virtual Chassis configuration consists of up to 10 switches.
Dimensions (W x H x D)	17.4 x 1.7 x 16.4 in. (44.21 x 4.32 x 41.73 cm.)
One rack unit (single switch)	
Backplane speed	128 Gbps (Virtual Chassis)
Data rate	136 Gbps
Aggregate switching capacity	264 Gbps
10/100/1000BASE-T Port densities	48 per platform Up to 480 in a Virtual Chassis configuration
100BASEFX/1000BASE-X (SFP) port densities	4 per platform (by using an optional four-port GbE uplink module) Up to 40 in a Virtual Chassis configuration
10GBASE-X port densities	2 per switch (by using an optional two-port 10 GbE uplink module) Up to 20 in a Virtual Chassis configuration
Resiliency	Internal, hot-swappable redundant power supply; field-replaceable fan tray with three fans; GRES in a Virtual Chassis configuration

Feature	Specification
Power options	Dual load-sharing, hot-swappable internal redundant power supplies ac: 320 W, 600 W, and 930 W autosensing; 100 – 120 V/200 – 240 V
Operating system	Junos software from Juniper Networks
Quality of service (QoS) queues/ports	8
MAC addresses	24,000
Jumbo frames	9216 bytes
IPv4 unicast or multicast routes	10,000 or 2,000
Number of VLANs	4,096
ARP entries	16,000

Hardware

The IBM Ethernet Switch J48E requires the following hardware:

- Single rack unit-height device (1RU)
- 48 10/100/1000BASE-T front-panel ports
- Optional GbE or 10 GbE front-panel uplink modules for connecting to other switches or upstream devices
- Scalable to 480 server access ports with up to 20 10 GbE uplinks to the core
- Back-panel Virtual Chassis ports that interconnect up to 10 switches over 128 Gbps backplane
- Redundant, internal hot-swappable power supplies
- Hot-swappable fan tray with redundant blowers
- Dual Routing Engines with GRES
- Single management interface

2.1.2 IBM Ethernet Switch J08E and J16E

The IBM Ethernet Switch J08E and the IBM Ethernet Switch J16E are high-density, high-performance, and highly available modular switches. They are intended for the most demanding datacenter core environments.

The IBM Ethernet Switch J08E and IBM Ethernet Switch J16E deliver the performance, scalability, and high availability that is required to support high-density datacenter, cloud computing, and enterprise campus environments. The high-density, high-performance J08E and J16E are also used for aggregating access switches that are deployed in top-of-rack or end-of-row applications in a data center. These switches also support Gigabit Ethernet server access in end-of-row deployments in a data center. The J08E delivers up to 960 million packets per second (Mpps) of high-density, wire-speed 10 GbE performance. The J16E delivers approximately 1.9 billion packets per second (Bpps) of 10 GbE performance. Both systems are designed to provide sufficient capacity to support the most demanding datacenter networks.

To maximize investments, the J08E and J16E use common wire-speed line cards and power supplies, ensuring consistent performance across the entire product family. Both switches also run the same Junos operating system, which helps to reduce capital and operational expenses across the datacenter infrastructure.

Line cards

The J08E and J16E can accommodate various combinations of the following Ethernet line card options.

The Exx 48 Port 1 GbE RJ-45 line card (Figure 2-3) offers the following features:

- 48 RJ-45 10/100/1000BASE-T interfaces
- Line-rate for any packet size or type (64 - 9216 bytes)
- 48 Gbps, 71 million packets per second
- 10–25 micron (μ) port-to-port latency depending on packet size
- Eight queues, 42 MB buffer per port

Figure 2-3 Exx 48 Port 1GbE RJ-45 line card

The Exx 48 Port 100FX/1000B-X line card (Figure 2-4) offers the following features:

- 48 small form-factor pluggable (SFP) 100/1000BASE-X interfaces
- Line-rate for any packet size or type (64 - 9216 bytes)
- 48 Gbps, 71 million packets per second
- 10 - 25 μ port-to-port latency depending on packet size
- Eight queues, 42 MB buffer per port

Figure 2-4 Exx 48 Port 100FX/1000B-X line card

The Exx 8 Port 10 GbE line card (Figure 2-5) offers the following features:

- Eight SFP+ 10 GbE interfaces
- Line-rate for any packet size or type (64 - 9216 bytes)
- 80 Gbps, 119 million packets per second
- 8 - 15 μ port-to-port latency depending on packet size
- Eight queues, 512 MB buffer per port

Figure 2-5 Exx 8 Port 10 GbE line card

Fully configured, a single J08E can support up to 384 GbE or 64 10 GbE ports at wire speed for all packet sizes. The J16E can support twice that number, 768 GbE or 128 10 GbE wire-speed ports for all packet sizes. Up to three 14 rack-unit (RU) J08Es or two 21 RU J16Es can fit in a standard 42RU wiring rack. This combination enables up to 1,536 line-rate GbE or 256 line-rate 10 GbE ports in a single rack and delivers one of the highest port densities in the industry.

Common architectural components

The two j-type e-series modular switches share several distinct architectural elements and characteristics. The Routing Engines that are used by the switches run the Junos operating system, which processes all Layer 2 and Layer 3 protocols, and manages individual chassis components.

Both switches also feature a switch fabric that provides a central crossbar matrix through which all data traffic passes. The fabric can deliver up to 320 Gbps (full duplex) per slot. It enables scalable wire-rate performance on all ports for packets of any size. Plus it provides a built-in migration path to support 100 GbE deployments without requiring any changes to the network infrastructure.

In addition, the line cards used by both switches include custom-designed ASIC-based Packet Forwarding Engines that process network traffic at wire rates. They also include a line-card processor that ensures control plane scalability.

All major switch components are hot-swappable. Also, all central functions are available in redundant configurations designed to provide high operational availability by allowing continuous system operation during maintenance or repairs.

Common features

The IBM Ethernet Switch J08E and J16E have the following common features:

- High-performance 8-slot (J08E) and 16-slot (J16E) switches that support a data center and campus LAN core and aggregation layer deployments
- Scalable switch fabric that delivers up to 320 Gbps per slot
- The support of 48-port 10/100/1000BASE-T and 100BASE-FX/1000BASE-X line cards up to 384 (J08E) or 768 (J16E) GbE ports per chassis

- Eight-port 10GBASE-X line cards with SFP+ interfaces that deliver up to 64 (J08E) or 128 (J16E) 10 GbE ports per chassis
- A carrier-class architecture that includes redundant internal Routing Engines, switch fabrics, power, and cooling, which ensures uninterrupted forwarding and maximum availability
- All IBM j-type Ethernet switches and routers run the modular, fault-tolerant Junos operating system

J08E description

The IBM Ethernet Switch J08E features a passive backplane design that supports a capacity of up to 6.2 terabits per second (Tbps). It is a high-performance, high-density platform that reduces cost and complexity, while improving overall scalability and providing carrier-class reliability in the datacenter.

The J08E base configuration includes the following features:

- A side-mounted hot-swappable fan tray with variable-speed fans
- One Switch Fabric and Routing Engine (SRE) module
- One dedicated switch fabric module
- Two 2000 watt internal power supplies

An optional second SRE module for hot-standby resiliency and up to six power supplies can provide component redundancy.

The J08E SRE module, shown in Figure 2-6, performs two functions. It incorporates switch fabric, control plane, and management plane functionality into a single module. It includes an integrated Routing Engine that features a 1.2 GHz processor with 2 GB of DRAM and 2 GB of flash memory. The Routing Engine on the SRE module includes a central CPU that performs all system control functions and maintains the hardware forwarding table and routing protocol states for the switch.

Figure 2-6 Switch Fabric and Routing Engine

The switch fabric for the J08E (Figure 2-7) serves as the central non-blocking matrix through which all network traffic passes. It is distributed across the SRE modules and the dedicated switch fabric module. Working together, the SRE and switch fabric modules deliver the necessary switching capacity for the J08E.

Figure 2-7 Switch fabric module

Specifications

Table 2-3 lists the specifications for the IBM Ethernet Switch J08E.

Table 2-3 IBM Ethernet Switch J08E specifications

Feature	Specification
Form factor	Eight-slot modular chassis
Dimensions (W x H x D)	17.25 x 24.25 x 21 in (43.82 x 61.6 x 53 cm); 14 rack units
Backplane speed	Up to 320 Gbps (full duplex) per line card Up to 6.2 Tbps per chassis
Data rate	Exx 48 Port 1 GbE RJ-45 line card: 96 Gbps Exx 48 Port 100FX/1000B-X line card: 96 Gbps Exx 8 Port 10 GbE line card: 160 Gbps
Throughput	Up to 120 Mpps per line card (eight-port 10 GbE) Up to 960 Mpps per system
10/100/1000BASE-T port densities	48 per Exx 48 Port 1 GbE RJ-45 line card; 384 per chassis
100BASE-FX/1000BASE-X (SFP) port densities	48 per Exx 48 Port 100FX/1000B-X line card; 384 per chassis
10GBASE-X port densities	8 per Exx 8 Port 10 GbE line card; 64 per chassis
Resiliency	Six power supply bays for N+1 or N+N redundancy; field-replaceable fan trays; redundant SRE; GRES
Power options	ac, 1200 W and 2000 W autosensing; 100-120V/200-240V (up to 6000 W or 10,000 W per chassis)
Operating system	Junos software
QoS queues/port	8
MAC addresses	160,000
Jumbo frames	9216 bytes
IPv4 unicast or multicast routes	400,000/320,000

Feature	Specification
Number of VLANs	4,096
ARP Entries	100,000

J16E description

The IBM Ethernet Switch J16E delivers a backplane capacity of up to 12.4 Tbps. It is designed for core datacenter deployments and demanding cloud computing and Internet exchange environments.

The base J16E configuration includes the following features:

- Two side-mounted hot-swappable fan trays with variable-speed fans
- One Routing Engine module
- Eight dedicated switch fabric modules
- Two 3,000 watt power supplies

Redundant configurations include an optional second Routing Engine module to provide hot-standby resiliency and up to six power supplies.

The J16E Routing Engine module supports control and management plane functionality with an integrated Routing Engine featuring a 1.2 GHz processor with 2 GB of DRAM and 2 GB of flash memory. Similar to the J08E, the Routing Engine of the J16E includes a CPU that performs all system control functions and maintains the hardware forwarding table and routing protocol states for the switch.

The switch fabric for the J16E is spread across eight rear-accessible switch fabric modules. All eight switch fabric modules are always active. They enable the switch to support line-rate Layer 2 and Layer 3 switching on all ports for packets of any size.

Specifications

Table 2-4 lists the specifications for the IBM Ethernet Switch J16E.

Table 2-4 IBM Ethernet Switch J16E specifications

Feature	Specification
Form factor	16-slot modular chassis
Dimensions (W x H x D)	17.25 x 36.5 x 26.5 in. (43.9 x 92.7 x 67.4 cm.); 21 rack units
Backplane speed	Up to 320 Gbps (full duplex) per line card Up to 12.4 Tbps per chassis
Data rate	Exx 48 Port 1 GbE RJ-45 line card: 96 Gbps Exx 48 Port 100FX/1000B-X line card: 96 Gbps Exx 8 Port 10 GbE line card: 160 Gbps
Throughput	Up to 120 Mpps per line card (eight-port 10 GbE) Up to 1.92 Bpps per system
10/100/1000BASE-T port densities	48 per Exx 48 Port 1GbE RJ-45 line card; 768 per chassis
100BASE-FX/1000BASE-X (SFP) port densities	48 per Exx 48 Port 100FX/1000B-X line card; 768 per chassis
10GBASE-X port densities	8 per Exx 8 Port 10 GbE line card; 128 per chassis

Feature	Specification
Resiliency	Six power supply bays for N+1 or N+N redundancy; field-replaceable fan trays; redundant SRE; GRES
Power options	ac, 3000 W 200-240 V (up to 15,000 W per chassis)
Operating system	Junos software
QoS queues/port	8
MAC addresses	160,000
Jumbo frames	9216 bytes
IPv4 unicast or multicast routes	400,000/320,000
Number of VLANs	4,096
ARP entries	100,000

Hardware

The IBM Ethernet Switch J08E and J16E have the following hardware requirements:

- 14 (J08E) rack-unit (RU) and 21 (J16E) RU chassis
- 8 (J08E) and 16 (J16E) dedicated I/O slots
- 6.2 (J08E) and 12.4 (J16E) Tbps backplane capacity
- 320 Gbps (full duplex) per slot fabric capacity
- Full 10 GbE line-rate forwarding (even under failure conditions)
- Dedicated data, control, and management planes
- LCD panel for system monitoring
- Energy efficient: 200,000 packets per second per watt
- Up to six load-sharing power supplies
- 6,000 W (J08E) and 15,000 W (J16E) maximum power configurations
- Redundant fans and controllers
- Side-to-side airflow

2.2 IBM j-type m-series Ethernet routers

The IBM j-type m-series Ethernet router portfolio is made up of the following models:

- IBM Ethernet Router J02M
- IBM Ethernet Router J06M
- IBM Ethernet Router J11M

Figure 2-8 shows the IBM j-type m-series Ethernet router portfolio.

Figure 2-8 IBM j-type m-series Ethernet routers

If you are familiar with the Juniper Networks products, Table 2-5 provides a cross-reference between the IBM j-type m-series Ethernet routers and the models offered by Juniper Networks.

Table 2-5 Cross-reference of IBM j-type m-series Ethernet routers to Juniper Networks models

IBM description	IBM machine type and model	Juniper networks model
IBM Ethernet Router J02M	4274-M02	MX240
IBM Ethernet Router J06M	4274-M06	MX480
IBM Ethernet Router J11M	4274-M11	MX960

IBM j-type m-series Ethernet routers with powerful switching and security capabilities deliver the reliability and flexibility needed to accelerate new business innovations. These routers offer innovations with advanced routing features, a single Junos operating system across all members of the j-type family, and high performance ASICs.

Optimized for Ethernet, these routers can be used to bring high performance, scalability, and availability to aggregation and core deployments in a data center, and WAN Edges.

All three j-type m-series Ethernet routers address high performance networking requirements that benefit from advanced routing features. These features include network virtualization with

Multiprotocol Label Switching (MPLS), QoS, logical interface scalability, high availability (HA), and low-latency multicast.

The following sections provide information about the IBM Ethernet routers in depth.

2.2.1 Key components of IBM j-type m-series Ethernet routers

The Routing Engine, the switch control board, and dense port concentrators are key components of each IBM j-type m-series Ethernet router.

Routing Engine

Each Routing Engine is a PC-based platform that runs the Junos operating system. Software processes that run on the Routing Engine maintain the routing tables. They manage the routing protocols used on the router. They also provide the interface for system management and user access to the router. They also control the router interfaces and some chassis components. Routing Engines communicate with dense port concentrators through dedicated out-of-band management channels. They provide a clear distinction between the control and forwarding planes. The Routing Engine is mounted within the switch control board as shown in Figure 2-9.

Figure 2-9 Switch control board carrier and Routing Engine daughter card

Switch control board

The switch control board powers cards on and off, resets, controls startup, controls clocking, and controls systems functions. It also monitors fan speeds, board power status, the system front panel. Additionally, the switch control board monitors the control and status of the power distribution module.

Integrated into the switch control board is the switch fabric, which interconnects all of the dense port concentrators within the chassis. The switch control board also has the Routing

Engine installed directly in it. Figure 2-10 shows the switch control board, with a Routing Engine installed in it.

Figure 2-10 Switch control board with a Routing Engine installed

Dense port concentrators

Dense port concentrators are optimized for Ethernet density and support up to 40 GbE ports or four 10 GbE ports. The dense port concentrator assembly combines packet forwarding and Ethernet interfaces on a single board, with four 10-Gbps Packet Forwarding Engines (PFE). Each PFE consists of one I-chip ASIC for Layer 3 processing and one Layer 2 network processor. The dense port concentrators interface with the power supplies and switch control boards.

Dense port concentrator cards for the IBM j-type m-series Ethernet routers enable cost- and performance-optimized Ethernet services. Dense port concentrators are interchangeable across the IBM j-type m-series Ethernet router portfolio to reduce costs and increase flexibility.

Dense port concentrators come in various types to enable robust solution design. Some dense port concentrators can switch and route to allow for a mid-range solution that can operate as either a Layer 3 router or Layer 2 switch. Others include scaled-down switching and routing functions for cost-optimized solutions. Yet others offer enhanced queuing for high-performance solutions that can support per-VLAN queuing, up to 64,000 queues per card. With this ability, providers can deploy business Ethernet services with committed bandwidths and QoS characteristics, increasing revenue opportunities. Figure 2-11 on page 54 shows dense port concentrators.

Figure 2-11 IBM j-type m-series dense port concentrators

Ethernet services

Ethernet services ensure that the IBM j-type m-series Ethernet routers deliver cost savings without sacrificing performance, reliability, scalability, or functionality. They also provide both switching and high performance Ethernet routing functionality.

Ethernet-based services include virtual private networks (VPNs), point-to-point connectivity, high-speed Internet access, and video-based offerings, including the following examples:

- Virtual Private LAN Service (VPLS) for multipoint connectivity
- Existing support for VPLS
- Video distribution of Internet Protocol Television (IPTV) services
- Ethernet aggregation at the campus or enterprise edge

- Support for dense GbE and 10 GbE configurations
- Provision of full layer 3 support for Campus Edge requirements

High performance

With exceptional performance, IBM j-type m-series Ethernet routers satisfy critical applications. They include voice, video, and data, with packet forwarding up to 720 Mpps and throughput up to 960 Gbps.

Two-tier network architecture

The high-performance and high-density characteristics of these routers enables substantial cost savings and operational efficiencies using a two-tier architecture. This efficient architecture collapses the core and aggregation layers of a traditional three-tier architecture, which results in fewer required devices, simplified management, and reduced power, cooling, and space requirements.

Network virtualization with MPLS

Leading organizations have adopted virtualization with their servers or storage. Virtualizing the network extends the benefits of virtualization. With network virtualization using MPLS, customers can improve the following areas:

- Privacy, with network segmentation
- End-user experience, with traffic engineering and QoS
- Network resiliency, with such features as Fast Re-Route, which establishes a predefined policy to re-route traffic during an outage, or a feature such as Bidirectional Forward Detection (BFD) to detect outages

MPLS in the enterprise provides great flexibility and reliability for deploying important applications. It enables consolidation of disparate networks onto a single network. It also delivers control through traffic segmentation and provides resiliency with fast re-route and traffic engineering.

With the virtualization capabilities of MPLS, one physical network can be configured and operate as many separate virtual networks.

Enterprises frequently use network virtualization to provide logical separation between multiple elements in a common fabric (for example, separation between HR and finance departments). Corporate mergers and acquisitions frequently encounter cases where multiple vital networks contain overlapping address spaces. They cannot be combined without considerable disruption. With MPLS, multiple networks can be connected and isolated without disruption through use of virtualization techniques.

High availability

IBM Ethernet routers are designed to provide the highest level of redundancy and resiliency. Both hardware and software high availability features ensure that critical services and customers stay connected at all times.

Low-latency multicast

The low-latency multicasting feature enables efficient streaming of financial, video, media, and IPTV content. Network performance is optimized while decreasing the amount of flow replication. The result is an excellent end-user experience and bandwidth savings.

Junos operating system

IBM j-Type m-series Ethernet routers use the Junos operating system, which is an advanced network operating system that powers some of the largest and most complex networks in the world. By using a common network operating system, IBM j-type m-series Ethernet routers deliver a consistent implementation and operation of control-plane features across all models.

To maintain that consistency, Junos software adheres to a highly disciplined development process that operates a single source code. It employs a highly available modular architecture that prevents isolated failures from bringing an entire system down. These attributes are fundamental to the core value of the operating system. With these attributes, switches and routers powered by Junos software can be updated simultaneously with the same software release. Each new release of Junos software is a true super set of the previous version. Customers can deploy new releases of the operating system with confidence that all existing capabilities have been maintained and will operate in the same way.

Simple, nondisruptive deployment

All IBM j-type m-series Ethernet routers are powered by Junos software. Therefore, customers can take advantage of the latest Ethernet enhancements without the cost and risk associated with introducing a new operating system to the network. With familiarity, knowledge, and integration of Junos software in existing back-office systems, they can drive down capital expenditures while rapidly rolling out advanced Ethernet access networks and services.

Interface scalability

The IBM j-type m-series Ethernet router chassis scales in size with choices of 2, 6 or 11 slots that can be populated with line cards for access or network interfaces. With up to 11 line card slots, the IBM Ethernet Router J11M supports up to 44 10 GbE ports or 440 GbE ports.

Advanced packet processing performance

Each IBM j-type m-series Ethernet router slot provides line-rate 40 Gbps packet forwarding.

Advanced QoS

IBM j-type m-series Ethernet routers offer superior QoS at the interface level, which improves port density, can reduce costs, and enables the network to ensure that services receive the appropriate level of service regardless of traffic conditions. With this level of QoS, enterprises can deliver various Layer 2 and Layer 3 services over Ethernet, such as the following examples:

- VLAN/transparent LAN
- L2/L3 VPNs
- Voice over IP (VoIP), with guaranteed service-level agreements (SLAs)

Ethernet router VPNs

Junos software supports the following types of VPNs:

- Layer 2 VPN links a set of sites that share common routing information and whose connectivity is controlled by a collection of policies. A Layer 2 VPN is not aware of routes within a customer's network. It provides private links between the customer's sites over the existing public Internet backbone.
- Layer 3 VPN links a set of sites that share common routing information and whose connectivity is controlled by a collection of policies. A Layer 3 VPN is aware of routes within a customer's network, requiring more configuration than a Layer 2 VPN. The sites that make up a Layer 3 VPN are connected over an existing public Internet backbone.

Virtualization in IBM j-type m-series Ethernet routers

Implementing virtualization in the data center comes with numerous benefits. Virtualization is used to improve asset utilization, scalability, privacy, resiliency, and more. IBM j-type m-series Ethernet routers provide a myriad of virtualization features and technologies, as shown in Figure 2-12.

Figure 2-12 IBM j-type m-series virtualization features and usage for the data center

To address enterprise datacenter requirements for network, management, control, or service virtualization, these features can be used individually or in combination to complement one another. It is insufficient to have a myriad of features. However, these features must be implemented consistently in one operating system, across the IBM j-type routing platforms, on top of advanced ASICs, enabling a collapsed two-tier data center architecture.

Network virtualization

Network virtualization improves network utilization, scalability, and resiliency.

MPLS network virtualization

With MPLS network virtualization, the physical network can be configured and operated as many separate virtual networks. MPLS network virtualization has the following benefits:

- Cost savings
- Improved privacy through traffic segmentation
- Improved user experience with traffic engineering and QoS
- Improved network resiliency with functionality such as Fast Re-Route and Bidirectional Forward Detection

Management virtualization

Management virtualization improves routing utilization.

Logical routers

Logical routers segment a physical router into multiple independent routers that perform independent routing tasks. This technology virtualizes the configuration and operation of a physical router into subsets, for increased manageability and protection. Enterprise customers wanting increased privacy and security, for example, can assign departments to separate virtual routing instances.

Control virtualization

Control virtualization improves system level utilization and security.

Virtual router

Virtual router provides multiple routing instances for the same physical router. This functionality keeps routing instances separate. Hence, overlapping IP addresses can exist in the virtual router instances. The virtual router functionality is a subset of the logical router functionality because there is no separation of management of the different virtual routers.

Virtual router functionality-lite

Virtual router functionality-lite is a simpler version of the virtual router functionality. Each router participates in a virtual routing environment in a peer-based fashion. Although it is simple to deploy, it does not scale for some enterprises because every router must maintain a virtual router functionality routing instance.

Link aggregation

Link aggregation provides a mechanism for combining multiple physically separate Layer 2 links as a single logical link. It helps enterprise data centers scale more bandwidth than a single Ethernet link can provide. It also enables savings on the expense of a higher-speed Ethernet link. This technology can also help enterprise data centers provide redundant links for greater resiliency. Thus datacenter managers can incrementally scale their investments by increasing utilization of existing resources while deriving increased security.

Service virtualization

Service virtualization offers many flexible options for secure virtual connectivity:

- Layer 2 VPN

  Layer 2 VPN offers Layer 2 services, over MPLS, to build point-to-point connections that connect different sites. Layer 2 VPNs are used to transport Layer 2 packets, across MPLS networks, without any knowledge of Layer 3 information of the networks, in the VPN. With this technology, data centers transport their existing L2 services, such as ATM, over an IP/MPLS network, minimizing capital expenses. This technology can also be used to transport Ethernet, allowing increased scalability.

- VPLS

  VPLS provides Ethernet-based point-to-multipoint communication over IP/MPLS networks. It allows geographically dispersed datacenter LANs to connect to each other across an MPLS backbone while maintaining Layer 2 connectivity. That is, VPLS creates a virtual network giving perception (to the constituent nodes) that they are on the same Ethernet LAN. Therefore, VPLS can provide an efficient and cost-effective method for data migration across enterprise data centers.

- Layer 3 VPN

  Layer 3 VPN provides private links between data center sites that share common routing information. A Layer 3 VPN is aware of routes within the network the VPN interconnects. For example, by mapping Layer 3 VPNs to virtual security zones in an advanced firewall, such as the IBM j-type Ethernet appliances, customers can layer many security policies together selectively by traffic type.

Examples of common virtualization practices

Logical routers can provide individual business units with the perception that they are working on independent routers. They include the following benefits among others:

- Improve routing utilization
- Align virtual routing instances with business units.

MPLS can help organizations in the following ways:

- Improve privacy
- Guarantee QOS to critical applications, such as VOIP, financial data, or multicast video streaming
- Help customers to achieve exceptional network resiliency through Bidirectional Forward Detection and Fast Re-Route without the need for dedicated SONET links, reducing capital and operating costs

VPLS can help datacenter managers migrate data between specific servers in co-located data centers. Enterprises no longer need dedicated Layer 2 links between the data centers.

The ability to selectively apply VPLS to specific VLANs is crucial to most enterprises that are interested in migrating only specific application data and not all the data in the data center. This ability enables greater scalability of the datacenter infrastructure.

Common features

The IBM j-type m-series Ethernet routers have the following features:

- High performance Ethernet L2, L2VPN, L3, or L3VPN router
- Delivers advanced routing features, including network virtualization with MPLS, low latency multicast, and QoS, without compromising performance
- Redundant hardware and high availability features that ensure that the network is always up and running:
  - Redundant hardware: cooling, power supplies, Routing Engines, and switch control boards
  - Modular operating system
  - Separate data and control planes
  - Graceful restart
  - Non-stop routing
  - MPLS Fast Reroute (FRR)
  - VPLS multi-homing
- Excellent performance to ensure that services and customers stay connected:
  - Powered by custom-designed ASICs
  - Enhanced QoS capabilities
  - Additional packet processing flexibility
  - Scaling enhancements including route lookup, next hop, logical interface scaling, and interface accounting
  - Additional control storage capacity to provide headroom for future features of the Junos operating system
  - Enhanced multicast
- Operational efficiencies and simplicity delivered through the Junos operating system
- Each chassis scaling in size with choices of 2, 6, or 11 slots that can be populated with line cards for access or network interfaces
- Flexible port densities and WAN interfaces (dense port-count line cards)
- Simultaneous support for Layer 2 and Layer 3 Ethernet services

- Deployment in two-tier collapsed architectures, which can yield operational savings through reduced power, cooling, and space requirements
- Target deployments: Ethernet router, with switching and security for datacenter aggregation and core, or WAN Edge
- Multicast: 256k multicast groups
- IP Version: IPv4, IPv6
- High availability: ISSU, NSR, MPLS FRR, VPLS multi-homing
- VPNs: MPLS L3 VPN, MPLS L2 VPN, IPSec VPN
- QoS: Layer 2 QoS and Layer 3 QoS
- Advanced services: Stateful firewall, IPSec, DAA
- Operating system: Junos software 9.6 R2 or higher

Model features

Table 2-6 are the individual features of each IBM j-type routing product.

Table 2-6 Model features of the IBM j-type routing platforms

Model features	J02M	J06M	J11M
Throughput	240 Gbps	480 Gpbs	960 Gbps
Dense port concentrator slots	2	6	11
Packet forwarding capacity	180 Mpps	360 Mpps	720 Mpps

Summary of hardware features

In summary, IBM j-type m-series Ethernet routers have the following features:

- Systems with 2, 6, and 11 slots with full duplex throughput per slot providing versatile port densities and WAN interfaces
- The ability for each chassis to scale in size and be populated with line cards for access or network interfaces
- Ideal fit for high-density deployments including the data center, campus, and WAN gateways
- Flexible architecture with IPv6, IPv4, industry leading IP/MPLS, and VPLS
- Powered by the I-chip ASIC, the Ethernet Router family includes features such as enhanced QoS capabilities and scaling enhancements and advanced packet processing performance — each slot provides line-rate 40 1-Gbps packet forwarding
- Simultaneous support for Layer 2 and Layer 3 Ethernet services: VPLS, RFC 2547bis IP/MPLS VPNs, Triple Play services ensures service flexibility
- Advanced and scalable Layer 2 services include VLAN tagging, MRP, VSRP, RSTP, and MSTP
- MEF 9 and MEF 14 certified
- Industry-leading MPLS services, which include IP over MPLS, Virtual Leased Line (VLL), Virtual Private LAN Service (VPLS), BGP/MPLS VPN
- Full suite of unicast and multicast IPv4 and IPv6 routing protocols include RIP, OSPG, BGP, ISIS and Protocol Independent Multicast (PIM)
- Comprehensive MPLS signaling and path-calculation algorithms for both traffic-engineered and non-traffic-engineered applications

- Secure multi-VRF routing for supporting virtual routing applications over non-MPLS backbones
- Scalability
  - 4 million Routing Information Base (RIB)/1 Million Forwarding Information Base (FIB) up to 6K BGP peers
  - 256K multicast groups
  - 64K VLANs and up to 1 million MAC addresses

2.3 Junos operating system

Junos software is the field-proven *common network operating system* that powers IBM j-type products in the data center. It enables the consolidation of switching and routing onto a common operating system, with feature consistency and interoperability across the entire datacenter network. Manageability and flexibility of the data center are enhanced to address business needs as they arise and improve datacenter operations. With a common set of tools, administrators can monitor, administer, and troubleshoot the network. In doing so, datacenter operations teams can function more efficiently with less training while providing higher availability for users. Unlike any other networking infrastructure operating system on the market, Junos provides one operating system. It is enhanced through a single release train and developed on a common modular architecture, giving enterprises a “1:1:1” advantage.

Figure 2-13 illustrates how Junos software runs on the entire IBM j-type infrastructure.

Figure 2-13 Junos software running on all IBM j-type systems

IBM provides high-performance network devices that create a responsive and trusted environment for accelerating the deployment of services and applications over a single network. Junos software is the foundation of these high-performance networks.

Unlike other complex, monolithic software architectures, Junos software incorporates key design and developmental differences to deliver increased network availability, operational efficiency, and flexibility. This approach has the following key advantages:

- One operating system
- One software release
- One modular software architecture

See *IBM j-type Ethernet Appliance Implementation*, SG24-7883, for information about the IBM j-type Ethernet Appliances shown in Figure 2-13.

The following sections examine each of these key advantages a little closer.

One operating system

Unlike other network operating systems that share a common name but splinter into many different programs, the Junos operating system is a single, cohesive operating system that is shared across all IBM j-type products. With this operating system, IBM engineers can develop software features once and share these features across all IBM j-type products simultaneously. Because features are common to a single source, they generally are implemented the same way for all IBM j-type products, thus reducing the training required to learn different tools and methods for each product. Because all IBM j-type products use the same code base, interoperability between products is not an issue.

One software release

Each new version of Junos software is released concurrently for all IBM j-type products following a preset quarterly schedule. Furthermore, each new version of software must include all working features released in previous releases of the software. They must not have any critical regression errors. This discipline ensures reliable operations for the entire release.

One modular software architecture

Although individual modules of the Junos software communicate through well-defined interfaces, each module runs in its own protected memory space, preventing one module from disrupting another. This separation enables the independent restart of each module as necessary. It is in contrast to monolithic operating systems, where a malfunction in one module can ripple to other modules and cause a full system crash or restart. This modular architecture then provides for high performance, high availability, device scalability, and security not found in other operating systems.

The Junos software is preinstalled on your IBM j-type equipment when you receive it from the factory. Thus, when you first power on the system, all software starts automatically. You need only to configure the software so that the system can participate in the network. You can upgrade the Junos software as new features are added or software problems are fixed. You normally obtain new software by downloading the software installation packages from the IBM Support Web page onto your system or onto another system on your local network. You then install the software upgrade onto the IBM j-type system. IBM j-type systems run only binary files supplied by IBM.

Each Junos software image includes a digitally signed manifest of executable programs that are registered with the system only if the signature can be validated. Junos software does not run any binary file without a registered signature. This feature protects the system against unauthorized software and activity that might compromise the integrity of your IBM j-type system.

For additional information about the Junos operating system, see Chapter 4, “Fundamental concepts of the Junos operating system” on page 107.

2.4 Simplification of the IBM Data Center Network solution

Today's data center consists of multiple platforms, each with a unique management application and interface. The management, configuration, and troubleshooting of the array of devices are complex and time consuming. Also, consider the increased cost of the administrators supporting these devices.

The network infrastructure is complex, typically consisting of multiple layers of devices, with an overlay of security that is configured separately at each layer. As a result of the network simplification, the number of devices has been reduced and associated power has been reduced. Also the space taken by devices has been reduced, and cooling requirements have been reduced, resulting in significant operational expenditure (OPEX) savings.

With IBM Data Center Network infrastructure solutions, enterprise data centers benefit from the following advantages:

- Reduced complexity

  All j-type routing and switching platforms use the same instance of the Junos operating system software for simpler feature deployments and upgrades. The virtualization of security services results in few physical devices to manage, thus preventing datacenter sprawl.

- Fewer layers of connectivity

  The IBM j-type e-series Ethernet switches with Virtual Chassis technology greatly simplify the data center. This technology reduces switch ports, links, switches, and aggregation layers, while improving performance, resiliency, and availability, managed space, and power costs. By simplifying the architecture and decreasing the number of devices, businesses can reduce power, space, and cooling expenses to create a greener, more energy-efficient data center.

- Support for high performance and high resiliency

  Using the top-of-rack j-type e-series Ethernet switches with Virtual Chassis technology and the j-type s-series Ethernet appliance, organizations can scale performance and increase resiliency while reducing the amount of equipment in the data center.

- Network security services

  The IBM j-type s-series Ethernet appliance delivers integrated services with scalable performance to consolidate network security by requiring fewer devices and providing centralized policy control and visibility to improve operational efficiency in the data center.

IBM high-performance network solutions for datacenter networks simplify the network design, security, and management of the data center, as shown in Figure 2-14 on page 64.

Figure 2-14 IBM Data Center Network solution simplification

The key criteria for IBM Data Center Network simplification consists of the following areas:

- Consolidation of the network layers:
  - Simplifies the architecture
  - Reduces space, power, and cooling requirements
  - Eliminates the aggregation layer
  - Offers great scalability and features
  - Reduces latency
- Virtualization of the access layer and minimization of the number of devices to manage:
  - Require fewer switches to manage (up to one-tenth)
  - Reduce uplinks
  - Reduce latency across racks
  - Optimize space, power, and cooling
  - Offer end-of-row chassis features at top-of-rack economics
  - Provide deployment flexibility (end-of-row or top-of-rack)
- Single operating system instead of multiple operating systems for the entire network:
  - Use only one operating system.
  - Install only one release.
  - Must understand only one architecture.
  - Feature parity provided and reduced certification time for new releases.

For more information about the IBM Data Center Network solution simplification, see *IBM j-type Data Center Networking Introduction*, SG24-7820.

2.5 More information

For a complete introduction to the IBM j-type portfolio of products and technologies, see *IBM j-type Data Center Networking Introduction*, SG24-7820, or the "IBM j-type Ethernet switches and routers" web page at the following address:

http://www.ibm.com/systems/networking/hardware/j-type/index.html

You can find all of the following documents on the IBM support site at this address:

http://www.ibm.com/systems/support/supportsite.wss/brandmain?brandind=5375876

For more information about the IBM j-type Ethernet switches and the related hardware, see the following documentation:

- *IBM Ethernet Switch J48E Complete Hardware Guide,* GA32-0663
- *IBM Ethernet Switch J48E Quick Start,* GA32-0664
- *IBM Ethernet Switch J08E Complete Hardware Guide,* GA32-0665
- *IBM Ethernet Switch J08E Quick Start,* GA32-0666
- *IBM Ethernet Switch J16E Complete Hardware Guide,* GA32-0667
- *IBM Ethernet Switch J16E Quick Start,* GA32-0668

For more information about the IBM j-type Ethernet routers the related hardware, see the following documentation:

- *IBM j-type m-series Ethernet Router Dense Port Concentrators (DPC) Guide,* GA32-0661
- *IBM j-type m-series Ethernet Routing Engine Installation Instructions,* GA32-0682
- *IBM j-type m-series Ethernet Services PIC Guide,* GA32-0662
- *IBM Ethernet Router J02M Hardware Guide,* GA32-0655
- *IBM Ethernet Router J02M Quick Start,* GA32-0656
- *IBM Ethernet Router J06M Hardware Guide,* GA32-0657
- *IBM Ethernet Router J06M Quick Start,* GA32-0658
- *IBM Ethernet Router J11M Hardware Guide,* GA32-0659
- *IBM Ethernet Router J11M Quick Start,* GA32-0660
- *JUNOS Software IBM j-type m-series Ethernet Routers Solutions Guide,* GA32-0683

For more information about the Junos software, see the following documentation:

- *Juniper Web Device Manager for IBM j-type Ethernet Switches and Routers Interface User Guide,* GA32-0688
- *JUNOS Software Access Privilege Configuration Guide,* GA32-0696
- *JUNOS Software Broadband Subscriber Management Solutions Guide,* GA32-0709
- *JUNOS Software Class of Service Configuration Guide,* GA32-0738
- *JUNOS Software CLI User Guide,* GA32-0697
- *JUNOS Software Configuration and Diagnostic Automation Guide,* GA32-0679
- *JUNOS Software Ethernet Routing Engine Media Upgrade Kit,* GA32-0681
- *JUNOS Software Feature Guide,* GA32-0739
- *JUNOS Software Hierarchy and RFC Reference,* GA32-0712

- *JUNOS Software High Availability Configuration Guide,* GA32-0670
- *JUNOS Software IBM j-type m-series Ethernet Routers Layer 2 Configuration Guide,* GA32-0708
- *JUNOS Software Installation and Upgrade Guide,* GA32-0695
- *JUNOS Software Interfaces Command Reference,* GA32-0672
- *JUNOS Software JUNOScript API Guide,* GA32-0674
- *JUNOS Software MPLS Applications Configuration Guide,* GA32-0702
- *JUNOS Software Multicast Protocols Configuration Guide,* GA32-0703
- *JUNOS Software NETCONF API Guide,* GA32-0678
- *JUNOS Software Network Interfaces Configuration Guide,* GA32-0706
- *JUNOS Software Network Management Configuration Guide,* GA32-0698
- *JUNOS Software Policy Framework Configuration Guide,* GA32-0704
- *JUNOS Software Routing Protocols and Policies Command Reference,* GA32-0673
- *JUNOS Software Services Interfaces Configuration Guide,* GA32-0707
- *JUNOS Software Subscriber Access Configuration Guide,* GA32-0711
- *JUNOS Software System Basics and Services Command Reference,* GA32-0671
- *JUNOS Software System Log Messages Reference,* GA32-0675
- *JUNOS Software VPNs Configuration Guide,* GA32-0705
- *JUNOScope Software User Guide,* GA32-0670

3

Initial hardware planning of IBM j-type Ethernet switches and routers

This chapter expands on a few of the items associated with installing the IBM j-type Ethernet switches and routers. It provides information about such items as power requirements, airflow, racks, and cabling as they pertain to successfully installing IBM j-type Ethernet switches and routers. The areas of the installation that are explained in this chapter have a long-term effect on the reliability, resilience, and overall success of the installation.

For information about a particular hardware model and a step-by-step procedure for installing the system, see the appropriate documents listed in 3.5, “More information” on page 105.

This chapter includes the following topics:

- Power considerations
- Cooling system
- Racks
- Cabling

© Copyright IBM Corp. 2011. All rights reserved.

3.1 Power considerations

Several power options are available when the equivalent Juniper Networks model is purchased. However, IBM only sells IBM j-type Ethernet switches and routers with fully redundant power supply options. As a result, the selection of power options is simplified so that you only need to choose the power plug type that is required. Additionally, you do not need to work through specific power calculations for each j-type device configured, because all j-type equipment is fully fault tolerant even with a fully loaded system.

The following sections provide details about power for the IBM j-type Ethernet switches and routers.

3.1.1 Power specifications and requirements for IBM Ethernet Switch J48E

The power supply in an IBM Ethernet Switch J48E is a hot-removable and hot-insertable field-replaceable unit (FRU). You can install this FRU (Figure 3-1) on the rear panel without powering off the switch or disrupting the switching function.

Figure 3-1 320 W ac power supply in an IBM Ethernet Switch J48E

These switches have an internal redundant power supply, making the power supply fully redundant.

Wait time for powering on or off a switch: After powering on a switch, wait for at least 60 seconds before powering it off. After powering off a switch, wait for at least 60 seconds before powering it back on.

After a switch has been powered on, it can take up to 60 seconds for status indicators to indicate that the power supply is functioning normally. Such status indicators include LEDs on the power supply, show chassis command output, and messages on the LCD. Ignore error indicators that are displayed during the first 60 seconds.

The switches use power that provides two dc output voltages:

- 12 V for system and logic power
- 48–51 V (or higher, to compensate for voltage drops along the path from the power supplies to the RJ-45 connector) for Power over Ethernet (PoE) ports

The power cord retainer clips extend from the power supply by approximately 3 inches.

The ac power supply LEDs in IBM Ethernet Switch J48E

The ac power supply in an IBM Ethernet Switch J48E (Figure 3-1 on page 68) is on the rear panel. Table 3-1 describes the LEDs on the ac power supplies.

Table 3-1 The ac power supply LEDs in an IBM Ethernet Switch J48E

LED	State and description
AC OK	Off—Disconnected from power or power is not coming into the power supply. On—Power is coming into the power supply.
DC OK	Off—Power supply is not sending out power correctly. On—Power supply is sending out power correctly.

Unlit AC and DC OK LED lights: If the AC OK LED and the DC OK LED are unlit, either the ac power cord is installed improperly or the power supply fuse has failed. If the AC OK LED is lit and the DC OK LED is unlit, the ac power supply is not installed properly or the power supply has an internal failure.

The ac power connection and power cord specifications for IBM Ethernet Switch J48E

Detachable ac power cords are supplied with the switch. The appliance coupler at the female end of the cord inserts into the ac appliance inlet on the faceplate of the ac power supply. The coupler is type C13 as described by International Electrotechnical Commission (IEC) standard 60320. The plug at the male end of the power cord fits into the power source outlet that is standard for your geographical location.

Table 3-2 lists ac power cord specifications provided for each country or region.

Table 3-2 The ac power cord specifications

Country or region	Electrical specifications	Plug standards
Australia	250 VAC, 10 A, 50 Hz	AS/NZ 3112–1993
China	250 VAC, 10 A, 50 Hz	GB2099.1 1996 and GB1002 1996 (CH1-10P)
Europe (except Italy and United Kingdom)	250 VAC, 10 A, 50 Hz	CEE (7)VII
Italy	250 VAC, 12 A, 50 Hz	CEI 23-16/VII
Japan	125 VAC, 16 A, 50 Hz or 60 Hz	JIS 8303
North America	125 VAC, 13 A, 60 Hz	NEMA 5-15
United Kingdom	250 VAC, 10 A, 50 Hz	BS 1363/A

Figure 3-2 illustrates the plug on the power cord for each country or region listed in Table 3-2 on page 69.

Figure 3-2 The ac plug types

3.1.2 Power specifications and requirements for IBM Ethernet Switches J08E and J16E

The ac power supply in IBM e-series modular switches is a hot-insertable and hot-removable FRU.

Six ac power supplies are installed in the switches. Power supplies are installed at the bottom of the chassis in slots PSU 0 through PSU 5 (left to right). All power supplies are accessible from the front of the chassis.

CAUTION:

The switch is pluggable type A equipment installed in a restricted-access location. It has a separate protective earthing terminal provided on the chassis in addition to the grounding pin of the power supply cord. This separate protective earthing terminal must be permanently connected to earth ground.

Each ac power supply weighs approximately 7 lb (3.2 kg) and has an independent 16 A rated ac inlet on its front. Each inlet requires a dedicated ac power feed. Each ac power supply has an Enable switch, a fan, and three LEDs on the faceplate that indicate the status of the power supply (Figure 3-3).

Figure 3-3 IBM j-type e-series ac power supply

Table 3-3 provides the status information for the LEDs of the power supply.

Table 3-3 Power supply LEDs on IBM e-series modular switches

LED	State	Description
Input OK	Unlit	Indicates one of the following statuses: ► The power supply is disconnected from the ac power feed. ► The ac power input voltage is not within the normal operating range. ► No ac power input is available.
	Green	The ac power input voltage is high line (200–240 VAC).
	Yellow This LED state applies to 2000 W ac power supplies only.	The ac power input voltage is low line (100–120 VAC).

LED	State	Description
Output OK	Unlit	Indicates one of the following statuses: ► The dc output voltage is not within normal operating range. ► Power supply is not supplying dc power correctly.
	Green	The dc power output is within the normal operating range.
	Yellow	The power supply has been disabled internally by the system.
Fail	Unlit	The power supply is functioning normally.
	Yellow	► On steadily; the power supply has failed. ► Blinking; the demand for output power exceeds the supply.

Each ac power supply comes with a power retainer that holds the power cord in place (Figure 3-4). The power retainer has a clip and an adjustment nut. The L-shaped ends of the clip hook into the bracket holes on each side of the ac appliance inlet on the faceplate. The adjustment nut holds the power cord in the correct position. For instructions to install the power retainer, see the appropriate hardware guide listed in 3.5, “More information” on page 105.

Figure 3-4 Power retainer for an ac power supply

Each power supply connects to the backplane in an IBM j-type e-series Ethernet switch. The backplane and the mid-plane distribute the output power produced by the power supplies to different switch components. Each ac power supply provides power to all the components in the switch.

Each power supply has its own fan and is cooled by its own internal cooling system. The airflow is from the front of the power supply to the back. Hot air exhausts from the rear of the chassis.

Airflow: The IBM J08E and J16E switches both have side-to-side airflow for the switch itself. Only the main power supplies have front-to-back airflow.

The ac power cord specifications for an IBM e-series modular switch

Each ac power supply has a single ac appliance inlet on the power supply that requires a dedicated ac power feed. Most sites distribute power through a main conduit that leads to frame-mounted power distribution panels, one of which can be at the top of the rack that houses the switch. An ac power cord connects each power supply to the power distribution panel.

Each detachable ac power cord is 2.5 meters (approximately 8 feet) long. The appliance coupler at the female end of the cord inserts into the ac appliance inlet on the faceplate of the ac power supply. The coupler type is C19 as described by International Electrotechnical Commission (IEC) standard 60320. The plug at the male end of the power cord fits into the power source outlet that is standard for your geographical location.

Guidelines for North America: In North America, ac power cords must not exceed 4.5 meters (approximately 15 feet) in length, to comply with National Electrical Code (NEC) Sections 400-8 (NFPA 75, 5-2.2) and 210-52 and Canadian Electrical Code (CEC) Section 4-010(3). The cords shipped with the switch are in compliance.

Table 3-4 lists the ac power cord specifications, by country or region, for an IBM e-series modular switch.

Table 3-4 The ac power cord specifications for an IBM e-series modular switch

Country or region	Electrical specifications	Plug standards
Australia	250 VAC, 16 A, 50 Hz	SAA/3/15
China	250 VAC, 16 A, 50 Hz	GB 1002-1996
Europe (except Denmark, Italy, Switzerland, and United Kingdom)	250 VAC, 16 A, 50 Hz	CEE (7)VII
Italy	250 VAC, 16 A, 50 Hz	CEI 23-16
Japan	250 VAC, 16 A, 50 Hz	NEMA 6–20 NEMA L6–20
North America	250 VAC, 16 A, 50 Hz	NEMA 6–20 NEMA L6–20
	125 VAC, 15 A, 50 Hz	NEMA 5–15 **Note:** Only for use with 2000 W ac power supply.
	125 VAC, 20 A, 50 Hz	NEMA 5–20 **Note:** Only for use with 2000 W ac power supply.
United Kingdom	250 VAC, 13 A, 50 Hz	BS 1363/A

Figure 3-5 shows the plug on the power cord for each country and region.

Figure 3-5 The ac plug types

CAUTION:

The ac power cord for an IBM e-series modular switch is intended for use with the switch only and not for any other use.

The ac power circuit breaker requirements for IBM j-type Ethernet switches

Each ac power supply has a single ac appliance inlet in the chassis directly above the power supply that requires a dedicated ac power feed. Use a dedicated customer site circuit breaker rated for 15 A (250 VAC) minimum for each ac power supply, or as required by local code.

Grounding cable and lug specifications for IBM e-series modular switches

CAUTION:

For installations that require a separate grounding conductor to the chassis, use the protective earthing terminal on the IBM e-series modular switch chassis to connect to earth ground. Before switch installation begins, a licensed electrician must attach a cable lug to the grounding cables that you supply.

For installations that require a separate grounding conductor to the chassis, the switch must be adequately grounded before power is connected to ensure proper operation and to meet safety and electromagnetic interference (EMI) requirements. To ground a switch, connect a grounding cable to earth ground, and then attach it to the chassis grounding points.

Two pairs of threaded inserts (Power Entry Module (PEM) nuts) are provided on the chassis for connecting the switch to earth ground. The first pair is on the right side toward the top rear corner of the chassis. The second pair is on the rear of chassis toward the right bottom corner of the chassis. Both pairs of grounding points fit UNC ¼-20 screws. The grounding points are spaced at 0.625 in. (15.86 mm).

Figure 3-6 shows the grounding lug. Two grounding lugs are shipped with the chassis.

Figure 3-6 J08E and J16E grounding lug

Earthing terminals: An IBM Ethernet Switch J16E has two protective earthing terminals provided on the chassis. Only one of these protective earthing terminals must be permanently connected to the earth.

Grounding is provided to an ac-powered switch when you plug its power supplies into grounded ac power receptacles.

The grounding cable that you provide for the switch must be 2 AWG (33.6 mm2), minimum 60°C wire, or as permitted by the local code.

CAUTION:

The switch is pluggable type A equipment installed in a restricted-access location. It has a separate protective earthing terminal provided on the chassis in addition to the grounding pin of the power supply cord. This separate protective earthing terminal must be permanently connected to earth ground for installations that require a separate grounding conductor to the chassis.

3.1.3 Power specifications and requirements for IBM Ethernet Routers J02M, J06M, and J11M

The IBM Ethernet j-type routers use ac power supplies. The power supplies connect to the mid plane, which distributes the different output voltages produced by the power supplies to the router components, depending on their voltage requirements. Each power supply is cooled by its own internal cooling system.

Redundant power supplies are hot removable and hot insertable. The J02M and J06M Ethernet j-type routers both have two power supplies that provide full redundancy. The J11M Ethernet j-type router has four power supplies and provides full redundancy. Routers that are configured with two power supplies are shipped with a blank panel installed over the power supply slots that are not populated.

Requirements for the ac power circuit breaker for IBM j-type Ethernet routers

Each ac power supply has a single ac appliance inlet in the chassis directly above the power supply that requires a dedicated ac power feed. Use a dedicated customer site circuit breaker rated for 15 A (250 VAC) minimum for each ac power supply, or as required by local code.

J02M and J06M ac power supply

Each ac power supply weighs approximately 5.0 lb (2.3 kg) and consists of one ac appliance inlet, one ac input switch, a fan, and LEDs to monitor the status of the power supply. Figure 3-7 shows the power supply used in the J02M and J06M. Each inlet requires a dedicated ac power feed and a dedicated customer site circuit breaker.

Figure 3-7 J02M and J06M ac power supply

J02M and J06M ac power supply LEDs

Each ac power supply faceplate contains three LEDs that indicate the status of the power supply. Table 3-5 provides information about the meaning of the various LED indicators.

Table 3-5 J02M and J06M ac Power Supply LEDs

Label	Color	State	Description
AC OK	Yellow	Off	The ac power input voltage is below 78 VAC.
	Green	On	The ac power input voltage is within 78–264 VAC.
DC OK	Green	Off	The dc power outputs generated by the power supply are not within the normal operating ranges.
		On	The dc power outputs generated by the power supply are within the normal operating ranges.
PS FAIL	Red	Off	The power supply is functioning normally.
		On	The power supply is not functioning normally, and its output voltage is out of regulation limits. Check AC OK and DC OK LEDs for more information.

Specifications for the ac power cord for J02M and J06M routers

Each ac power supply has a single ac appliance inlet on the power supply that requires a dedicated ac power feed. Most sites distribute power through a main conduit that leads to frame-mounted power distribution panels, one of which can be at the top of the rack that houses the router. An ac power cord connects each power supply to the power distribution panel.

Detachable ac power cords, each 2.5 m (approximately 8 ft) long, are supplied with the router. The C19 appliance coupler at the female end of the cord inserts into the ac appliance inlet coupler, type C20 (right angle) as described by IEC standard 60320. The plug at the male end of the power cord fits into the power source receptacle that is standard for your geographical location.

Table 3-6 provides specifications for the ac power cord provided for each country or region.

Table 3-6 Specifications for the ac power cord for J02M and J06M routers

Country or region	Electrical specifications	Plug standards
Australia	240 VAC, 50 Hz ac	SAA/3/15
China	220 VAC, 50 Hz ac	GH2-16P
Europe (except Denmark, Italy, Switzerland, and United Kingdom)	220 or 230 VAC, 50 Hz ac	CEE 7/7
Italy	230 VAC, 50 Hz ac	CEI 23-16/VII
Japan	125 VAC 50 or 60 Hz ac 220 VAC, 50 or 60 Hz ac	JIS 8303 NEMA 6–20P
North America	125 VAC 60 Hz ac 125 VAC 60 Hz ac 250 VAC, 50 or 60 Hz ac 250 VAC, 50 or 60 Hz ac	NEMA 5-15P NEMA L5-15P NEMA 6–20 NEMA L6–20P
United Kingdom	240 VAC, 50 Hz ac	BS89/13

Figure 3-8 shows the plug on the ac power cord provided for each country or region.

Figure 3-8 The ac cord plug types for the J02M and J06M routers

J11M ac power supply

The J11M router contains four ac power supplies (Figure 3-9) at the rear of the chassis in slots PEM0 through PEM3 (left to right).

Each ac power supply provides power to all components in the router. Four ac power supplies provide full power redundancy. If one power supply fails or is removed, the remaining power supplies instantly assume the entire electrical load without interruption. Three power supplies provide the maximum configuration with full power while the router is operational. Each ac power supply has a corresponding ac appliance inlet in the chassis directly above the power supply. Each inlet requires a dedicated ac power feed and a dedicated 15 A (250 VAC) circuit breaker.

Figure 3-9 J11M ac power supply

J11M ac power supply LEDs

Each ac power supply faceplate contains three LEDs that indicate the status of the power supply as shown in Table 3-7.

Table 3-7 J11M ac power supply LEDs

Label	Color	State	Description
AC OK	Green	Off	The ac power applied to the power supply is not within the normal operating range.
		On	The ac power applied to power supply is within the normal operating range.
DC OK	Green	Off	The dc power outputs generated by the power supply are not within the normal operating ranges.
		On	The dc power outputs generated by the power supply are within the normal operating ranges.
PS FAIL	Red	Off	The power supply is functioning normally.
		On	The power supply is not functioning normally, and its output voltage is out of regulation limits. Check the AC OK and DC OK LEDs for more information.

The ac power cord specifications for the J02M and J06M routers

Each ac power supply has a single ac appliance inlet on the power supply that requires a dedicated ac power feed. Most sites distribute power through a main conduit that leads to frame-mounted power distribution panels. One of these panels can be at the top of the rack that houses the router. An ac power cord connects each power supply to the power distribution panel.

Detachable ac power cords, each 2.5 m (approximately 8 ft.) long, are supplied with the router. The C19 appliance coupler at the female end of the cord inserts into the ac appliance inlet coupler, type C20 (right angle) as described by IEC standard 60320. The plug at the male end of the power cord fits into the power source receptacle that is standard for your geographical location.

Table 3-8 provides specifications for the ac power cord provided for each country or region.

Table 3-8 The ac cord specifications for the J11M router

Country or region	Electrical specifications	Plug standards
Australia	240 VAC, 50 Hz ac	SAA/3
China	220 VAC, 50 Hz ac	PSB-10
Europe (except Denmark, Italy, Switzerland, and United Kingdom)	220 or 230 VAC, 50 Hz ac	CEE 7/7
Italy	230 VAC, 50 Hz ac	CEI 23-16/VII
Japan	220 VAC, 50 or 60 Hz ac	NEMA 6–20P
North America	250 VAC, 50 or 60 Hz ac	NEMA L6–20P
United Kingdom	240 VAC, 50 Hz ac	BS89/13

Figure 3-10 depicts the plug on the ac power cord for each country or region.

Figure 3-10 The ac cord plug types for the J11M router

CAUTION:

The ac power cord for an IBM e-series modular switch is intended for use with the switch only and not for any other use.

J11M chassis grounding specifications

To meet safety and EMI requirements and to ensure proper operation, the router must be adequately grounded before power is connected. To ground the routers, you must connect a grounding cable to earth ground and then attach it to the chassis grounding points using the two screws that are provided. Two threaded inserts (PEM nuts) are provided on the right of the lower rear of the chassis for connecting the router to earth ground as shown in Figure 3-11.

Figure 3-11 Grounding lug for the J11M

CAUTION:

The switch is pluggable type A equipment installed in a restricted-access location. It has a separate protective earthing terminal provided on the chassis in addition to the grounding pin of the power supply cord. This separate protective earthing terminal must be permanently connected to earth ground.

3.2 Cooling system

The following sections provide information about the cooling systems for IBM j-type Ethernet switches and routers.

3.2.1 IBM Ethernet Switch J48E

The cooling system in an IBM Ethernet Switch J48E consists of an FRU fan tray with three fans as shown in Figure 3-12.

Figure 3-12 J48E Fan Tray FRU

The fan tray is at the rear of the chassis and provides side-to-rear chassis cooling as shown in Figure 3-13.

Figure 3-13 J48E airflow

The fan tray used in the switch comes with load-sharing redundancy that can tolerate a single fan failure at room temperature (below 45° C/113° F) to still provide sufficient cooling.

Temperature sensors in the chassis monitor the temperature within the chassis. The system raises an alarm if the fan fails or if the temperature inside the chassis rises above permitted levels. If the temperature inside the chassis rises above the threshold, the system shuts down automatically and the temperature shutdown LED on the rear panel is lit.

3.2.2 IBM Ethernet Switches J08E and J16E

The cooling system in the IBM Ethernet Switch J08E and J16E consists of a single fan tray. The fan tray is a hot-insertable and hot-removable FRU. A handle on the front faceplate facilitates handling of the fan tray. The fan tray contains 12 fans as shown in Figure 3-14.

Figure 3-14 J08E and J16E fan tray

Both fan trays install vertically on the front of the chassis, one on the top left and the other on the bottom left. The top and bottom fan trays are identical and interchangeable. Both fan trays can be removed and replaced from the front of the chassis. The switch continues to operate for a limited time (2 minutes) after a fan tray has been removed.

Replacing the fan tray: You must replace the fan tray within 2 minutes of removing it or the system will shut down.

The switch has side-to-side airflow in the front of the chassis. The air intake to cool the chassis from the mid plane to the chassis front is on the right side of the chassis. Air is pulled into the chassis and is pushed through the line card cage toward the fan trays. Hot air exhausts from the left side of the chassis.

Figure 3-15 shows how the air flows through the switches. The air intake to cool the power supplies is in the front of each power supply unit. The exhaust for the hot air collected from the power supplies is on the rear of the chassis at the bottom. See the side view in Figure 3-15 for this airflow.

Figure 3-15 J08E and J16E airflow

Cooling for the rear of the chassis and the switch fabric modules is done with front-to-side airflow. The air intake to cool the chassis from the midplane to the chassis rear is on the front of the chassis, just below the slots for the power supplies. Air is pulled in from the chassis front toward the chassis rear and then pulled to the top of the chassis by the top fan tray. The hot air is forced to turn left and exhausts from the left side of the chassis as shown in Figure 3-16 on page 87.

The Routing Engine module monitors the temperature of switch components. Under normal operating conditions, the fans in the fan trays run at less than full speed. Each fan tray has two fan tray controllers.

In each fan tray, the fans are numbered 1 through 9. Fans 1 through 5 are controlled by the first fan tray controller. Fans 6 through 9 are controlled by the second fan tray controller. If one fan tray controller fails, the other fan tray controller keeps the remaining fans in the fan tray working. This way, the switch can continue to operate normally while the remaining working fans cool the chassis sufficiently.

If the ambient temperature rises above the threshold 113°F (45°C), the speed of the working fans is automatically adjusted to keep the temperature within the acceptable range, 32°F (0°C) through 104°F (40°C).

The fan trays continue to operate indefinitely and provide sufficient cooling even when a single fan fails, if the room temperature is within the operating range.

Figure 3-16 J08E and J16E air exhaust

CAUTION:

Do not block the air intake below the power supply slots.

3.2.3 IBM Ethernet Routers J02M, J06M, and J11M

The cooling system consists of the following components:

- Fan tray
- Air filter

The J02M and J06M cooling system

The cooling system components work together to keep all router components within the acceptable temperature range. Figure 3-17 illustrates the airflow of the J02M cooling system.

Figure 3-17 Airflow of the J02M cooling system

Figure 3-18 illustrates the airflow of the J06M cooling system.

Figure 3-18 Airflow of the J06M cooling system

The routers have one fan tray and one air filter that installs vertically in the rear of the router. The fan tray contains three or six fans depending on system model. The air intake to cool the chassis is on the side of the chassis next to the air filter. Air is pulled through the chassis toward the fan tray, where it is exhausted out the side of the system. The air intake to cool the power supplies is in the front of the router above the craft interface. The exhaust for the power supplies is on the rear bulkhead power supplies.

Additionally, as shown in Figure 3-19 and Figure 3-20, the J02M and J06M also have a filter that requires occasional cleaning.

Figure 3-19 Fan tray and air filter for the J02M cooling system

Figure 3-20 Fan tray and filter for the J06M cooling system

The host subsystem monitors the temperature of the router components. When the router is operating normally, the fans function at lower than full speed. If a fan fails or the ambient temperature rises above a threshold, the speed of the remaining fans is automatically adjusted to keep the temperature within the acceptable range. If the ambient maximum temperature specification is exceeded and the system cannot be adequately cooled, the Routing Engine shuts down the system by disabling output power from each PEM.

Maintaining the J02M and J06M cooling system components

Regularly inspect the air filter. A dirty air filter restricts airflow in the unit, producing a negative effect on the ventilation of the chassis. The filter degrades over time. Periodically replace the filter in use and spares. Replace the filter every six months.

Monitor the status of the fans. A fan tray contains multiple fans that work in unison to cool the router components. If one fan fails, the host subsystem adjusts the speed of the remaining fans to maintain proper cooling.

J11M cooling system

The cooling system consists of the following components:

- Upper front fan tray
- Lower front fan tray
- Front air filter

The cooling system components work together to keep all router components within the acceptable temperature range. Figure 3-21 shows three views of the airflow of the J11M router.

Figure 3-21 Airflow through the J11M router

The router has two fan trays in the front of the router that install horizontally above and below the card cage. Each fan tray contains six fans. The fan trays are interchangeable and are hot-insertable and hot-removable (Figure 3-22).

Figure 3-22 J11M fan tray

Additionally, as shown in Figure 3-23, the J11M also has a pair of filters and filter trays that require occasional cleaning.

Figure 3-23 Air filter and air filter tray for the J11M

The host subsystem monitors the temperature of the router components. When the router is operating normally, the fans function at lower than full speed. If a fan fails or the ambient temperature rises above a threshold, the speed of the remaining fans is automatically adjusted to keep the temperature within the acceptable range. If the ambient maximum temperature specification is exceeded and the system cannot be adequately cooled, the Routing Engine shuts down the system by disabling output power from each PEM.

A single air intake is in the front of the router. Air is pushed up through the dense port concentrator card cage and through the upper fan tray. There it combines in a common exhaust plenum and is exhausted out the upper rear of the system.

Maintaining the J11M cooling system components

Regularly inspect the air filter. A dirty air filter restricts airflow in the unit, producing a negative effect on the ventilation of the chassis. The filter degrades over time. Periodically replace the filter in use and the spares. Replace the filter every 6 months.

Monitor the status of the fans. A fan tray contains multiple fans that work in unison to cool the router components. If one fan fails, the host subsystem adjusts the speed of the remaining fans to maintain proper cooling. A red alarm is triggered when a fan fails, and a yellow alarm and red alarm are triggered when a fan tray is removed.

3.3 Racks

Various racks are used for the IBM j-type Ethernet switches and routers as explained in the following sections.

3.3.1 IBM Ethernet Switch J48E

You can mount the IBM Ethernet Switch J48E units in four-post open or Telco racks. Two-post open rack installations are not supported. For additional information about the open rack requirements, see the appropriate hardware guide listed in the 3.5, “More information” on page 105.

If you are installing multiple switches to function as a Virtual Chassis, you must install the switches in a rack or cabinet. One pair of mounting brackets, for mounting the switch on two posts, is supplied with each switch. You can order a four-post rack-mount kit separately.

Additionally, the IBM Ethernet Switch J48E can be mounted in a cabinet that contains a 19-in. rack as defined in *Cabinets, Racks, Panels, and Associated Equipment* (document number EIA-310-D). This document is published by the Electronics Industry Association, which you can find at the following address:

http://www.eia.org

The cabinet has the following requirements:

- Cabinet size and clearance
- Cabinet airflow requirements

Table 3-9 summarizes the cabinet requirements and specifications for IBM Ethernet Switch J48E.

Table 3-9 Cabinet requirements and specifications for IBM Ethernet Switch J48E

Cabinet requirement	Guidelines for IBM Ethernet Switch J48E
Cabinet size and clearance	► The minimum cabinet size for accommodating an IBM Ethernet Switch J48E is 23.62 in. (60 cm) wide and 30.0 in. (76.2 cm) deep. Large cabinets improve airflow and reduce the chance of overheating. To accommodate a single switch in a two-post or four-post rack, the cabinet must be at least 1 U high. A U is the standard rack unit defined in *Cabinets, Racks, Panels, and Associated Equipment* (document number EIA-310–D) published by the Electronics Industry Association. ► With adequate cooling air and airflow clearance, you can stack up to 42 IBM Ethernet Switch J48E units in a cabinet with a two-post or four-post rack that has at least 42 U of usable vertical space. The rack must meet the strength requirements to support the weight. ► The minimum total clearance inside the cabinet is 29.2 in. (74.17 cm) between the inside of the front door and the inside of the rear door.
Cabinet airflow requirements	When you mount the switch in a cabinet, ensure that ventilation through the cabinet is sufficient to prevent overheating. Consider the following requirements list when planning for chassis cooling: ► Ensure that the cool air supply you provide through the cabinet adequately dissipates the thermal output of the switch (or switches). ► Ensure that the cabinet allows the chassis hot exhaust air to exit the cabinet without recirculating into the switch. An open cabinet (without a top or doors) that employs hot air exhaust extraction from the top allows the best airflow through the chassis. If the cabinet contains a top or doors, perforations in these elements assist with removing the hot air exhaust. For an illustration of chassis airflow, see Figure 3-25 on page 94. ► The switch fans exhaust hot air through the rear of the chassis. Install the switch in the cabinet in a way that maximizes the open space at the rear of the chassis to provide maximum clearance for critical airflow. ► Route and dress all cables to minimize the blockage of airflow to and from the chassis. ► Ensure that the spacing of rails and adjacent racks allows for the proper clearance around the switch and rack as specified in “Clearance requirements for airflow and hardware maintenance for an IBM Ethernet Switch J48E” on page 93.

Clearance requirements for airflow and hardware maintenance for an IBM Ethernet Switch J48E

When planning the site for installing an IBM Ethernet Switch J48E, you must allow sufficient clearance around the installed switch as shown in Figure 3-24.

Figure 3-24 Clearance requirements for airflow and hardware maintenance for IBM Ethernet Switch J48E

Cooling system clearance

Allow at least 6 in. (15.2 cm) of clearance on the side between devices that have fans or blowers installed. Allow 2.8 in. (7 cm) between the side of the chassis and any non-heat-producing surface such as a wall. For the cooling system to function properly, the airflow around the chassis must be unrestricted. Figure 3-25 on page 94 shows the airflow through the switch chassis.

You must ensure that the exhaust from other equipment does not blow into the intake vents of the chassis in the following situations:

- You are mounting the switch in a rack or cabinet with other equipment.
- You are placing it on the desktop or floor near other equipment.

Clearance for maintenance

Leave at least 24 in. (61 cm) both in front of and behind the switch. For service personnel to remove and install hardware components, you must leave adequate space at the front and back of the switch. NEBS GR-63 recommends that you allow at least 30 in. (76.2 cm) in front of the rack or cabinet and 24 in. (61 cm) behind the rack or cabinet.

Figure 3-25 Airflow through the IBM Ethernet Switch J48E chassis

3.3.2 IBM Ethernet Switches J08E and J16E

You can mount the IBM Ethernet Switch J08E or J16E units in four-post open or Telco racks. Two-post open rack installations are not supported. For additional information about the open rack requirements, see the appropriate hardware guide listed in 3.5, “More information” on page 105.

Additionally, the IBM Ethernet Switch J08E and J16E can be mounted in a cabinet that contains a 19-in. rack as defined in *Cabinets, Racks, Panels, and Associated Equipment* (document number EIA-310-D) published by the Electronics Industry Association.

The cabinet has the following requirements:

- Cabinet size and clearance
- Cabinet airflow requirements

Table 3-10 on page 95 summarizes the cabinet requirements and specifications for IBM e-series modular switches.

Installation: The switch can be installed only in a four-post rack or cabinet. Installation in a two-post rack or cabinet is not supported.

Table 3-10 Cabinet requirements and specifications for the IBM e-series modular switches

Cabinet requirements	Guidelines for the IBM e-series modular switches
Cabinet size and clearance	► The minimum cabinet size for accommodating the switch is 23.62 in. (60 cm) wide and 36.0 in. (91.4 cm) deep. Large cabinets improve airflow and reduce the chance of overheating. To accommodate a single switch in a four-post rack, the cabinet must be at least 21 U high (or 22 U if you install the power cord tray, which is optional for the four-post rack). A U is the standard rack unit defined in *Cabinets, Racks, Panels, and Associated Equipment* (document number EIA-310–D) published by the Electronics Industry Association. ► With adequate cooling air and airflow clearance, you can stack two J16E switches, or three J08E switches, in a cabinet with a four-post rack that has at least 42 U of usable vertical space. (44 U or 45 U are required if you install the optional power cord trays.) In all cases, the rack must meet the strength requirements to support the weight of the installed switches.
Cabinet airflow requirements	When you mount the switch in a cabinet, ensure that ventilation through the cabinet is sufficient to prevent overheating. Consider the following requirements list when planning for chassis cooling: ► Ensure that the cool air supply you provide through the cabinet adequately dissipates the thermal output of the switch (or switches). ► Ensure that the cabinet allows the chassis hot exhaust air to exit the cabinet without recirculating into the switch. An open cabinet (without a top or doors) that employs hot air exhaust extraction from the top allows the best airflow through the chassis. If the cabinet contains a top or doors, perforations in these elements assist with removing the hot air exhaust. ► The switch fans exhaust hot air through the right side of the chassis (the left side when you face the front of the chassis, where the fan tray slides in). Install the switch in the cabinet in a way that maximizes the open space on the fan tray side of the chassis. This maximizes the clearance for critical airflow. ► Route and dress all cables to minimize the blockage of airflow to and from the chassis. ► Ensure that the spacing of rails and adjacent racks allows for the proper clearance around the switch and rack as specified in “Clearance requirements for airflow and hardware maintenance for the IBM j-type e-series modular switches”

Clearance requirements for airflow and hardware maintenance for the IBM j-type e-series modular switches

When planning the site for installing an IBM j-type e-series modular switches, you must allow sufficient clearance around the switches.

Cooling system clearance

Allow at least 6 in. (15.2 cm) of clearance on each side of the chassis. For the cooling system to function properly, the airflow around the chassis must be unrestricted. See Figure 3-15 on page 86 and Figure 3-16 on page 87.

If you are mounting the switches in a rack or cabinet along with other equipment, ensure that the exhaust from other equipment does not blow into the intake vents of the IBM e-series modular switches.

Clearance for maintenance

Leave at least 24 in. (61 cm) both in front of and behind the switches. Allow at least 6 in. (15.2 cm) of clearance on each side of the chassis.

Leave adequate space at the front of the switches for service personnel to remove and install hardware components. NEBS GR-63 recommends that you allow at least 30 in. (76.2 cm) in front of the rack or cabinet and 24 in. (61 cm) behind the rack or cabinet.

3.3.3 IBM Ethernet Routers J02M, J06M, and J11M

You can mount the IBM Ethernet Routers in four-post open or Telco racks. Two-post open rack installations are not supported. For more information about the open rack requirements, see the appropriate hardware guide listed in 3.5, “More information” on page 105.

Additionally, the IBM Ethernet Routers can be mounted in a cabinet that contains a 19-in. rack as defined in *Cabinets, Racks, Panels, and Associated Equipment* (document number EIA-310-D) published by the Electronics Industry Association.

The cabinet has the following requirements:

- Cabinet size and clearance
- Cabinet airflow requirements

Table 3-11 summarizes cabinet requirements and specifications for IBM e-series modular switches.

Installation: The switch can be installed only in a four-post rack or cabinet. Installation in a two-post rack or cabinet is not supported.

Table 3-11 Cabinet requirements and specifications for an IBM m-series Ethernet Router

Cabinet requirements	Guidelines for the IBM m-series Ethernet Router
Cabinet size and clearance	► The minimum cabinet size that can accommodate the routers is 482-mm wide and 800-mm deep. A cabinet larger than the minimum requirement provides better airflow and reduces the chance of overheating. To accommodate a single router, the cabinet must be at least J02M and J06M = 13 U high, or J11M = 16 U high. A U is the standard rack unit defined in *Cabinets, Racks, Panels, and Associated Equipment* (document number EIA-310–D) published by the Electronics Industry Association. ► If you provide adequate cooling air and airflow clearance in a cabinet that has at least 42 U of usable vertical space, you can stack two IBM Ethernet Router J11M units with the standard cable manger installed, four IBM Ethernet Router J06M units with the standard cable manager installed, or seven IBM Ethernet Router J02M units with the standard cable manager installed. In all cases, the rack must meet the strength requirements to support the weight. ► The minimum front and rear clearance requirements depend on the mounting configuration you choose. The minimum total clearance inside the cabinet is 30.7 in. (780 mm) between the inside of the front door and the inside of the rear door.

Cabinet requirements	Guidelines for the IBM m-series Ethernet Router
Cabinet airflow requirements	When you install the router in a cabinet, you must ensure that ventilation through the cabinet is sufficient to prevent overheating. Consider the following requirements when planning for chassis cooling: ► Airflow must always be from front to back with respect to the rack. If the device has side-to-rear airflow, make provisions to ensure that fresh air from the front of the rack is supplied to the inlets, and exhaust exits the rear of the rack. The device must not interfere with the cooling of other systems in the rack. Fillers must be used as appropriate in the rack to ensure that heated exhaust air is not recirculated back to the front of the rack. Use care around cables to ensure no leakage of air in situations where recirculation might result. ► Ensure that the cool air supply you provide through the cabinet can adequately dissipate the thermal output of the routers. ► Ensure that the cabinet allows the chassis hot exhaust air to exit from the cabinet without recirculating into the routers. An open cabinet (without a top or doors) that employs hot air exhaust extraction from the top allows the best airflow through the chassis. If the cabinet contains a top or doors, perforations in these elements assist with removing the hot air exhaust. ► Route and dress all cables to minimize the blockage of airflow to and from the chassis. ► Ensure that the spacing of rails and adjacent racks allows for the proper clearance around the router and rack as specified in “Clearance requirements for airflow and hardware maintenance for the IBM j-type m-series Ethernet routers”.

Clearance requirements for airflow and hardware maintenance for the IBM j-type m-series Ethernet routers

When planning the site for installing an IBM j-type m-series Ethernet routers, you must allow sufficient clearance around the routers.

Cabinet airflow requirements for IBM j-type m-series Ethernet routers

Before you install the router in a cabinet, ensure that ventilation through the cabinet is sufficient to prevent overheating. Consider the following requirements when planning for chassis cooling:

- Airflow must always be from front to back with respect to the rack. If the device has side-to-rear airflow, make provisions to ensure that fresh air from the front of the rack is supplied to the inlets and exhaust exits the rear of the rack. The device must not interfere with the cooling of other systems in the rack. Fillers must be used as appropriate in the rack to ensure that heated exhaust air is not recirculated back to the front of the rack. Use care around cables to ensure no leakage of air in situations where recirculation might result.
- Ensure that the cool air supply that you provide through the cabinet can adequately dissipate the thermal output of the router.
- Ensure that the cabinet allows the chassis hot exhaust air to exit from the cabinet without recirculating onto the router. An open cabinet (without a top or doors) that employs hot air exhaust extraction from the top allows the best airflow through the chassis. If the cabinet contains a top or doors, perforations in these elements assist with removing the hot air exhaust. For an illustration of chassis airflow, see Figure 3-26 on page 98 for the J02M, Figure 3-27 on page 98 for the J06M, and Figure 3-28 on page 98 for the J11M.
- Install the router as close as possible to the front of the cabinet so that the cable manager just clears the inside of the front door. This placement maximizes the clearance in the rear of the cabinet for critical airflow.
- Route and dress all cables to minimize the blockage of airflow to and from the chassis.

Figure 3-26 Airflow of the J02M

Figure 3-27 Airflow of the J06M

Figure 3-28 Airflow of the J11M

Clearance for maintenance

Leave at least 24 in. (61 cm) in front of and behind the routers. Allow at least 6 in. (15.2 cm) of clearance on each side of the chassis.

Leave adequate space at the front of the switches for service personnel to remove and install hardware components. NEBS GR-63 recommends that you allow at least 30 in. (76.2 cm) in front of the rack or cabinet and 24 in. (61 cm) behind the rack or cabinet.

3.4 Cabling

This section provides details about cabling for the IBM j-type Ethernet switches and routers. All IBM j-type Ethernet switches and routers use the same cable types for the interfaces. Therefore, we do not break this section into various equipment types, but rather discuss cable types that apply to the IBM j-type Ethernet switches and routers.

3.4.1 Cables connecting to management devices

Two basic management ports are possible, a console port and a management port. The management port might be labeled as an Ethernet port on some models of the IBM j-type equipment. However, the labeling difference does not affect the cable used for this out-of-band management port.

Console port

The cable used for the console port is an RS-232 (EIA-232) serial cable with an RJ45 on one end and a DB9 on the other end. The ports are configured as data terminal equipment (DTE). Table 3-12 details the pinout for the DB9 end.

Table 3-12 Console cable pinout for the DB9 connector

PIN	Signal	Description
1	DCD	Carrier detect
2	RxD	Receive data
3	TxD	Transmit data
4	DTR	Data terminal ready
5	Ground	Signal ground
6	DSR	Data set ready
7	RTS	Ready to send
8	CTS	Clear to send
9	Ring	Ring indicator

Table 3-13 details the pinout for the RJ45 end.

Table 3-13 Console cable pinout for the RJ45 connector

PIN	Signal	Description
1	RTS	Request to send
2	DTR	Data terminal ready
3	TxD	Transmit data
4	Ground	Signal ground
5	Ground	Signal ground
6	RxD	Receive data
7	DSR/DCD	Data set ready
8	CTS	Clear to send

Aux port: The port labeled Aux has the same cable requirements as the console port.

Management (MGMT or Ethernet) port

In addition to the console and Aux ports that require an RS-232 cable for connecting a console directly to the system, an out-of-band management port is available. Depending on the j-type system in use, this port is labeled either Management (MGMT) or Ethernet.

This port is an auto-sensing 10/100-Mbps Ethernet RJ-45 receptacle that accepts a standard Ethernet cable for connecting the system to a management LAN (or other device that supports out-of-band management). Table 3-14 describes the RJ-45 connector pinout.

Table 3-14 RJ-45 connector pinout for the Ethernet port

PIN	Signal
1	TX +
2	TX -
3	RX +
4	Termination network
5	Termination network
6	RX -
7	Termination network
8	Termination network

Standard RJ45 Ethernet ports

Excluding the Management (MGMT or Ethernet) ports used for out-of-band management, the IBM j-type systems provide RJ45 ports for Ethernet connectivity to LAN devices. These ports are auto-sensing and can accommodate several different interface options. Fortunately, these different interface options can all use the same cable. However, the pins carry different signals depending on what interface option is in use.

10/100 Ethernet cables

The pinout for 10/100 Ethernet is the same as the pinout listed in Table 3-14 on page 100 for the management port.

Power over Ethernet

PoE uses the same Cat-5 cable as 10/100 Ethernet, but it employs the unused lines to deliver power to an end device. Table 3-15 provides the pinout for PoE with an RJ45 connector.

Table 3-15 Pinout for PoE

PIN	Signal name	Description
1	TD +	Transmit data
2	TD -	Transmit data
3	RD +	Receive data
4	Positive Vport	PoE
5	Positive Vport	PoE
6	RD -	Receive data
7	Negative Vport	PoE
8	Negative Vport	PoE

Ethernet Bus 1000BaseT

In addition to 10/100 Ethernet and PoE, the standard RJ45 Ethernet ports also supports Gigabit Ethernet (GbE) using a Cat-5 cable. However, the pins of the cable are used in a differently than either of the other interfaces. Table 3-16 describes the Ethernet 1000BaseT (twisted pair pinout).

Table 3-16 Ethernet Bus 1000BaseT RJ45 pinout

PIN	Signal name	Description
1	BI_DA+	Bi-directional pair +A
2	BI_DA-	Bi-directional pair -A
3	BI_DB+	Bi-directional pair +B
4	BI_DB-	Bi-directional pair -B
5	BI_DC+	Bi-directional pair +C
6	BI_DC-	Bi-directional pair -C
7	BI_DD+	Bi-directional pair +D
8	BI_DD-	Bi-directional pair -D

Optical ports

Throughout the IBM j-type Ethernet switches and routers, optical ports are used for various types of connectivity, such as the following examples:

- 100Base-FX
- 1000Base-SX
- 1000Base-LX
- 1000Base-LH (or 1000Base-ZX)

- 10GBase-SR
- 10GBase-LR
- 10GBase-LRM
- Virtual Chassis Interconnect

All optical ports use standard small form-factor pluggable (SFP), or 10 gigabit small form-factor pluggable (XFP) transceivers that require Lucent Connectors (LCs).

Optical cables

Fiber-optic cables used with j-type equipment must match the specifications of the transceiver that is used for that particular port. Certain details, such as the mode (single mode, or multimode), must be correct for the exact transceiver. Other details, such as size (50 micron or 62.5 micron), must be sufficient for the particular connection, such as the distance that the signal must traverse.

Optical cable maintenance

Unlike copper cables, fiber optic cables require special attention and maintenance.

To maintain fiber-optic cables, follow these guidelines:

- When you unplug a fiber-optic cable from a transceiver, place rubber safety caps over the transceiver and on the end of the cable.
- Anchor fiber-optic cable to avoid stress on the connectors. When attaching a fiber-optic cable to a transceiver, secure the fiber-optic cable so that it is not supporting its own weight as it hangs to the floor. Never let a fiber-optic cable hang free from the connector.
- Avoid bending fiber-optic cables beyond their minimum bend radius. Bending fiber-optic cables into arcs smaller than a few inches in diameter can damage the cables and cause problems that are difficult to diagnose.
- Frequent plugging and unplugging of fiber-optic cables in and out of optical instruments can damage the instruments, which are expensive to repair. Attach a short fiber extension to the optical equipment. Any wear and tear due to frequent plugging and unplugging is then absorbed by the short fiber extension, which is easier and less expensive to replace than the instruments.
- Keep the fiber-optic cable connections clean. Microdeposits of oil and dust in the canal of the transceiver or cable connector can cause a loss of light, reduction in signal power, and possibly intermittent problems with the optical connection.
- To clean the transceiver canal, use an appropriate fiber-cleaning device such as Fiber Optic Adaptor Cleaning Wands (RIFOCS). Follow the directions in the cleaning kit that you use. After cleaning the transceiver, ensure that the connector tip of the fiber-optic cable is clean. Use only an approved alcohol-free fiber-optic cable cleaning kit, such as the Opptex Cletop-S Fiber Cleaner.

Virtual Chassis ports connector pinout information for IBM Ethernet Switch J48E

All of the other cable considerations that have already been presented are generic across the entire IBM j-type product line. However, the Virtual Chassis ports (VCPs) connector is only available on the IBM Ethernet Switch J48E.

On the J48E, you can configure a Virtual Chassis by using fiber-optic cables, which were explained earlier in this chapter. Alternatively, you can use a 68-pin connector cable to interconnect switches to form a Virtual Chassis. The cable is provided with the switch.

Table 3-17 provides the connector pinout information for the VCPs.

Table 3-17 VCPs connector pinout

Pin number	Pin name
A1	GND
A2	P1TXP0
A3	P1TXN0
A4	GND
A5	P1TXP1
A6	P1TXN1
A7	GND
A8	P1TXP2
A9	P1TXN2
A10	GND
A11	P1TXP3
A12	P1TXN3
A13	GND
A14	NC
A15	NC
A16	GND
A17	NC
A18	NC
A19	NC
A20	NC
A21	NC
A22	GND
A23	P2TXP0
A24	P2TXN0
A25	GND
A26	P2TXP1
A27	P2TXN1
A28	GND
A29	P2TXP2
A30	P2TXN2
A31	GND
A32	P2TXP3

Pin number	Pin name
A33	P2TXN3
A34	GND
B1	GND
B2	P1RXP0
B3	P1RXN0
B4	GND
B5	P1RXP1
B6	P1RXN1
B7	GND
B8	P1RXP2
B9	1RXN2
B10	GND
B11	P1RXP3
B12	P1RXN3
B13	GND
B14	NC
B15	NC
B16	NC
B17	NC
B18	NC
B19	NC
B20	NC
B21	NC
B22	GND
B23	P2RXP0
B24	P2RXN0
B25	GND
B26	P2RXP1
B27	P2RXN1
B28	GND
B29	P2RXP2
B30	P2RXN2
B31	GND
B32	P2RXP3

Pin number	Pin name
B33	P2RXN3
B34	GND

Alternately, an IBM Ethernet Switch J48E has an FRU uplink module on the front panel.

3.5 More information

All of the following documents are available on the IBM support site at this address:

http://www.ibm.com/systems/support/supportsite.wss/brandmain?brandind=5375876

The installation of each model of the IBM j-type e-series and m-series systems is documented in both the quick start and hardware guides for each system type. For additional information and step-by-step installation procedures, see the following appropriate documentation:

- *IBM Ethernet Switch J48E Complete Hardware Guide,* GA32-0663
- *IBM Ethernet Switch J48E Quick Start,* GA32-0664
- *IBM Ethernet Switch J08E Complete Hardware Guide,* GA32-0665
- *IBM Ethernet Switch J08E Quick Start,* GA32-0666
- *IBM Ethernet Switch J16E Complete Hardware Guide,* GA32-0667
- *IBM Ethernet Switch J16E Quick Start,* GA32-0668

For more information about the IBM j-type Ethernet routers the related hardware, see the following documentation:

- *IBM j-type m-series Ethernet Router Dense Port Concentrators (DPC) Guide,* GA32-0661
- *IBM j-type m-series Ethernet Routing Engine Installation Instructions,* GA32-0682
- *IBM j-type m-series Ethernet Services PIC Guide,* GA32-0662
- *IBM Ethernet Router J02M Hardware Guide,* GA32-0655
- *IBM Ethernet Router J02M Quick Start,* GA32-0656
- *IBM Ethernet Router J06M Hardware Guide,* GA32-0657
- *IBM Ethernet Router J06M Quick Start,* GA32-0658
- *IBM Ethernet Router J11M Hardware Guide,* GA32-0659
- *IBM Ethernet Router J11M Quick Start,* GA32-0660
- *JUNOS Software IBM j-type m-series Ethernet Routers Solutions Guide,* GA32-0683

For more information about Junos software, see the following documentation:

- *Juniper Web Device Manager for IBM j-type Ethernet Switches and Routers Interface User Guide,* GA32-0688
- *JUNOS Software Access Privilege Configuration Guide,* GA32-0696
- *JUNOS Software Broadband Subscriber Management Solutions Guide,* GA32-0709
- *JUNOS Software Class of Service Configuration Guide,* GA32-0738
- *JUNOS Software CLI User Guide,* GA32-0697

- *JUNOS Software Configuration and Diagnostic Automation Guide,* GA32-0679
- *JUNOS Software Ethernet Routing Engine Media Upgrade Kit,* GA32-0681
- *JUNOS Software Feature Guide,* GA32-0739
- *JUNOS Software Hierarchy and RFC Reference,* GA32-0712
- *JUNOS Software High Availability Configuration Guide,* GA32-0670
- *JUNOS Software IBM j-type m-series Ethernet Routers Layer 2 Configuration Guide,* GA32-0708
- *JUNOS Software Installation and Upgrade Guide,* GA32-0695
- *JUNOS Software Interfaces Command Reference,* GA32-0672
- *JUNOS Software JUNOScript API Guide,* GA32-0674
- *JUNOS Software MPLS Applications Configuration Guide,* GA32-0702
- *JUNOS Software Multicast Protocols Configuration Guide,* GA32-0703
- *JUNOS Software NETCONF API Guide,* GA32-0678
- *JUNOS Software Network Interfaces Configuration Guide,* GA32-0706
- *JUNOS Software Network Management Configuration Guide,* GA32-0698
- *JUNOS Software Policy Framework Configuration Guide,* GA32-0704
- *JUNOS Software Routing Protocols and Policies Command Reference,* GA32-0673
- *JUNOS Software Services Interfaces Configuration Guide,* GA32-0707
- *JUNOS Software Subscriber Access Configuration Guide,* GA32-0711
- *JUNOS Software System Basics and Services Command Reference,* GA32-0671
- *JUNOS Software System Log Messages Reference,* GA32-0675
- *JUNOS Software VPNs Configuration Guide,* GA32-0705
- *JUNOScope Software User Guide,* GA32-0670

4

Fundamental concepts of the Junos operating system

This chapter presents the main aspects of the Junos operating system. Each IBM j-type data center networking product runs a version of the Junos operating system that was created by Juniper Networks and is built into the software architecture. It provides carrier-class network software to highly available data centers of all sizes.

This chapter includes the following topics:

- Overview of Junos
- The Junos architecture
- Routing Engine
- Packet Forwarding Engine
- The Junos software configuration model
- Junos routing and forwarding tables
- User interface options
- Introduction to Junos CLI

© Copyright IBM Corp. 2011. All rights reserved.

4.1 Overview of Junos

Junos has been designed based on the following key points, as summarized in Figure 4-1:

- One operating system

  The most fundamental virtue of Junos software is a single source code base. With this code base, engineers can develop new features one time and then share the code, as applicable, across the many platforms running the Junos operating system.

  A single, cohesive operating system providing a consistent user experience makes planning easier, day-to-day operations more intuitive, and changes faster for customers. Administrators can configure and manage functionality from the basic chassis to complex routing using the same tools across devices to monitor, manage, and update the entire network.

- One software release train (schedule)

  The Junos approach to software development produces a stable code base. This code base reduces the number of unplanned system events and the time and trouble of planned maintenance and upgrades.

  New versions of Junos software are released in a defined quarterly schedule, providing stable and predicable delivery of new functionality.

- One modular software architecture

  Each module of the Junos software runs in its own protected memory space, preventing one module from disrupting another. This software architecture also enables the independent restart of each module as necessary. It provides for a high level of performance, high availability, security, and device scalability that is not found in other operating systems.

Figure 4-1 Overview of Junos

4.2 The Junos architecture

Junos-based platforms share a common design that separates the control and forwarding planes of a switch or router as shown in Figure 4-2. All IBM j-type family products consist of two major components:

- Routing Engine
- Packet Forwarding Engine

Figure 4-2 The Junos architecture

4.3 Routing Engine

The Routing Engine controls the routing updates and system management. The Routing Engine consists of routing protocol software processes running inside a protected memory environment on a general-purpose computer platform. The routing engine handles all the routing protocol processes and other software processes that control the routers' interfaces, some of the chassis components, system management, and user access to the router. These routers and software processes run on top of a kernel that interacts with the Packet Forwarding Engine.

4.3.1 Routing Engine components for Junos software

Junos software is based on the FreeBSD UNIX® operating system. The open source software has been modified and hardened to operate in the specialized environment of the router. For example, some executable programs were deleted, but other utilities were de-emphasized. Additionally, certain daemons were added to enhance the routing functionality. The result of this transformation is the kernel, which is the heart of Junos software. The kernel is responsible for operating multiple daemons that perform the actual

functions of the router. Each daemon operates in its own protected memory space, which is also controlled by the kernel. This separation provides isolation between the processes and resiliency in the event of a process failure. This isolation is important in a core routing platform because a single process failure does not cause the entire router to cease functioning.

The Junos software runs on the Routing Engine. As mentioned, the Junos software processes run on top of the kernel, which enables communication between processes and provides a direct link to the Packet Forwarding Engine software. The Junos software can be used to configure routing protocols and router interface properties and to monitor and troubleshoot protocol and network connectivity problems.

The following processes are among the most important Junos software processes:

- Routing Engine kernel

  The Routing Engine kernel provides the underlying infrastructure for all Junos software processes. It also provides the link between the routing tables and the forwarding table of the Routing Engine. It is also responsible for all communication with the Packet Forwarding Engine.

- Initialization process

  An initialization process starts and monitors all the other software processes when the router boots. If a software process terminates or fails to start, the initialization process attempts to restart it a limited number of times and logs any failure information for further investigation.

- Management process

  The management process manages the configuration of the router and all user commands. The management process is responsible for notifying other daemons when a new configuration is committed.

- Routing protocol process

  The routing protocol process controls the routing protocols that run on the router. This process starts all configured routing protocols and handles all routing messages. It maintains one or more routing tables, which consolidate the routing information learned from all routing protocols. From this routing information, the routing protocol process determines the active routes to network destinations and installs these routes into the forwarding table of the Routing Engine. Finally, it implements a routing policy that you can use to control the routing information that is transferred between the routing protocols and the routing table. You can also filter routing information using routing policy and set the properties associated with routes.

- Interface process

  With the interface process, you can configure and control the physical interface devices and logical interfaces that are present in a device. You can configure interface properties such as the interface location, the interface encapsulation, and interface-specific properties. You can configure the interfaces currently present in the router and interfaces that are not present but that you might add later. The Junos interface process communicates through the kernel with the interface process in the Packet Forwarding Engine. With this feature, you can track the status and condition of the interface of the router.

- Chassis process

  With the chassis process, you can configure and control the properties of the router, including conditions that trigger alarms. The chassis process on the Routing Engine communicates directly with its peer processes running on the Packet Forwarding Engine.

- SNMP process

  The Junos software supports the Simple Network Management Protocol (SNMP), which helps administrators monitor the state of a router. The SNMP software is controlled by the Junos SNMP and Management Information Base II processes, which consist of an SNMP master agent and various subagents.

4.4 Packet Forwarding Engine

The Packet Forwarding Engine is responsible for forwarding packets through the device. The Packet Forwarding Engine is implemented by using specific application-specific integrated circuits (ASICs). The separation between control operations, such as protocol updates and system management, and packet forwarding provides superior performance and reliable deterministic operation

The Packet Forwarding Engine receives the forwarding table and bridging table from the Routing Engine through an internal link. Forwarding table and bridging table updates are a high priority for the software kernel.

The Packet Forwarding Engine forwards packets based on the Layer 2 and Layer 3 forwarding tables provided by the Routing Engine. It also implements several advanced services. Examples of advanced services include policies that provide rate limiting, firewall filters, and class of service.

4.5 The Junos software configuration model

The router configuration is saved using a commit model (Figure 4-3), in which a candidate configuration is modified as desired and then committed to the system.

Figure 4-3 Commit model

4.5.1 Active configuration

Configuration changes to the Junos software do not take effect immediately. The active configuration is the configuration currently operational in the device.

4.5.2 Candidate configuration

The candidate configuration is a temporary configuration that might become the active configuration. Initially, Junos creates a candidate configuration and populates it with the active configuration of the switch. The user can modify this configuration and commit the changes. This action causes the candidate configuration to became the active configuration.

4.5.3 Configuration history

When committed, the router checks the configuration for syntax errors. If no errors are found, the configuration is saved as a `juniper.conf.gz` file and activated. The former active configuration file is saved as the first rollback configuration file (`juniper.conf.1.gz`), and all other rollback configuration files are incremented by 1. For example, `juniper.conf.1.gz` is incremented to `juniper.conf.2.gz`, making it the second rollback configuration file.

The router can have a maximum of 49 rollback configurations (1–49) saved on the system. On the router, the active configuration file and the first three rollback files (`juniper.conf.gz.1`, `juniper.conf.gz.2`, `juniper.conf.gz.3`) are in the `/config` directory. If the recommended `rescue.conf.gz` rescue file is saved on the system, this file must also be saved in the `/config` directory. The factory default files are in the `/etc/config` directory.

Two mechanisms are used to propagate the configurations between the Routing Engines within a router:

- Synchronization

  This mechanism propagates a configuration from one Routing Engine to a second Routing Engine within the same router chassis. To synchronize the configurations of a router, use the **`commit synchronize`** command-line interface (CLI) command. If one of the Routing Engines is locked, the synchronization fails. If synchronization fails because of a locked configuration file, you can use the **`commit synchronize force`** command. This command overrides the lock and synchronizes the configuration files.

- Distribution

  This mechanism propagates a configuration across the routing plane on a multichassis router. Distribution occurs automatically. No user command is available to control the distribution process. If a configuration is locked during a distribution of a configuration, the locked configuration does not receive the distributed configuration file. Therefore, the synchronization fails. You must clear the lock before the configuration and resynchronize the routing planes.

4.6 Junos routing and forwarding tables

A major function of the Junos routing protocol process is to maintain the Routing Engine's routing tables, and from these tables, to determine the active routes to network destinations. The routing protocol process then installs these routes into the forwarding table of the Routing Engine. The Junos kernel then copies this forwarding table to the Packet Forwarding Engine. The routing protocol process maintains multiple routing tables. By default, it maintains the following three routing tables. You can configure additional routing tables to suit your requirements.

- Unicast routing table

  This routing table stores routing information for all unicast routing protocols running on the router. BGP, IS-IS, OSPF, and RIP all store their routing information in this routing table. You can configure additional routes, such as static routes, to be included in this routing table. BGP, IS-IS, OSPF, and RIP use the routes in this routing table when advertising routing information to their neighbors.

- Multicast routing table

  This routing table stores routing information for all the running multicast protocols. DVMRP and Protocol Independent Multicast (PIM) both store their routing information in this routing table. You can configure additional routes to be included in this routing table.

- Multiprotocol Label Switching (MPLS) routing table

  This routing table stores MPLS path and label information. With each routing table, the routing protocol process uses the collected routing information to determine active routes to network destinations. For unicast routes, the routing protocol process determines active routes by choosing the most preferred route, which is the route with the lowest preference value. By default, the preference value of the route is a function of how the routing protocol process learned about the route. You can modify the default preference value by using the routing policy and the software configuration parameters.

  For multicast traffic, the routing protocol process determines active routes based on traffic flow and other parameters specified by the multicast routing protocol algorithms. The routing protocol process then installs one or more active routes to each network destination into the forwarding table of the Routing Engine.

4.7 User interface options

You can use any of the following methods to configure Junos software:

- Junos CLI

  The Junos CLI is a straightforward command interface. You use Emacs-style keyboard sequences to move around on a command line and scroll through a buffer that contains recently executed commands. You type commands on a single line and then press Enter to run the commands. The CLI also provides command help and command completion.

- ASCII File

  You can load an ASCII file that contains a router configuration that you created earlier, either on this system or another system. You can then activate and run the configuration file as is, or you can edit it using the CLI and then activate it.

- J-Web

  As an alternative to entering CLI commands, Junos software supports a J-Web graphical user interface (GUI). With the J-Web user interface, you can monitor, configure, troubleshoot, and manage the router on a client using a web browser with HTTP or HTTP over Secure Sockets Layer (HTTPS) enabled.

- JUNOScript API

  The JUNOScript API is an XML application that client applications use to request and change configuration information about IBM j-type Ethernet routers. This API is customized for Junos software, and operations in the API are equivalent to Junos CLI configuration mode commands. The JUNOScript API includes a set of Perl modules that enable client applications to communicate with a JUNOScript server on the router. The Perl modules are used to develop custom applications for configuring and monitoring Junos software.

- NETCONF API

  The NETCONF API is an XML application that client applications can use to request and change configuration information about IBM j-type Ethernet routers. This API is customized for Junos software and includes features that accommodate the configuration data models of multiple vendors. The NETCONF API includes a set of Perl modules that enable client applications to communicate with a NETCONF server on the router. The Perl modules are used to develop custom applications for configuring and monitoring Junos software.

- Configuration commit scripts

  You can create and use scripts that run at commit time to enforce custom configuration rules. If a configuration breaks the custom rules, the script can generate the following actions, among others that the Junos software performs:

  - Generating custom error messages
  - Generating custom warning messages
  - Generating custom system log messages
  - Changing the configuration

  With configuration commit scripts, you can also create macros. These macros expand simplified custom aliases for frequently used configuration statements into standard Junos configuration statements. Commit scripts are written in Extensible Stylesheet Language Transformations (XSLT).

4.8 Introduction to Junos CLI

Junos CLI is a specific command shell that runs on top of a UNIX-based operating system kernel. The CLI provides command help and command completion.

The CLI also provides various UNIX utilities, such as the following examples:

- Emacs-style keyboard sequences so that you can move around on a command line and scroll through recently executed commands
- Regular expression matching to locate and replace values and identifiers in a configuration, filter command output, or log file entries
- Store and archive router files on a UNIX-based file system
- Exit from the CLI environment and create a UNIX C shell or Bourne shell to navigate the file system
- Manage switch processes

The CLI has two modes, operational mode and configuration mode.

4.8.1 Operational mode

In operational mode, you enter commands to monitor and troubleshoot switch hardware and software and network connectivity. Operational mode is indicated by the > prompt, for example:

```
user@switch>
```

For more information about operational mode, see Chapter 6, “User interface” on page 143.

4.8.2 Configuration mode

In configuration mode, you can define all properties of the Junos software, including interfaces, VLANs, Virtual Chassis information, routing protocols, user access, and several system hardware properties.

To enter configuration mode, enter the **`configure`** command:

```
user@switch> configure
```

In configuration mode, you view and change the candidate configuration file. With candidate configuration, you can make configuration changes without causing operational changes to the current operating configuration, called the *active configuration*. When you commit the changes you added to the candidate configuration, the system updates the active configuration. By using candidate configurations, you can change your configuration without potentially damaging your current network operations.

For more information about configuration mode, see Chapter 6, “User interface” on page 143.

4.8.3 More information

For more information about Junos software and basic configuration, see the *System Basics Configuration Guide* at the following web address:

http://www.juniper.net/techpubs/en_US/junos10.1/information-products/topic-collections/config-guide-system-basics/config-guide-system-basics.pdf

For more information about Junos CLI, see the *CLI User Guide* at the following web address:

http://www.juniper.net/techpubs/en_US/junos10.1/information-products/topic-collections/swconfig-cli/swconfig-cli.pdf

5

Initial configuration

This chapter explains the initial setup that is required to implement the switches and routers. It provides information about the EZSetup script, which is a startup script that simplifies the initial setup and implementation of the switches and routers.

This chapter includes the following topics:

- Network topology
- Initial setup for the J08E, J16E, and J48E switches
- LCD panel and status LEDs
- Initial setup for the J02M, J06M, and J11M routers

© Copyright IBM Corp. 2011. All rights reserved.

5.1 Network topology

The physical layout of the equipment used for this IBM Redbooks publication includes the following equipment, which is shown in Figure 5-1:

- IBM Ethernet Switch J48E - 3 units
- IBM Ethernet Switch J08E - 1 unit
- IBM Ethernet Router J11M - 1 unit
- IBM Ethernet Router J02M - 1 unit
- IBM Ethernet Appliance J58S - 2 units

While this book only focuses on the IBM j-type Ethernet switches and the routers, all these devices were part of the configuration and testing of examples. See *IBM j-type Ethernet Appliance Implementation*, SG24-7883, for more information about the IBM j-type Ethernet Appliances shown in Figure 5-1 on page 118.

Figure 5-1 illustrates how each device cables with other devices in the lab environment. Three units of IBM Ethernet Switch J48Es are interconnected to build a single management switch called a *Virtual Chassis switch*. IBM Ethernet switches and routers are interconnected in full mesh topology. Connections of IBM Ethernet routers and appliances have a full mesh topology to provide change flexibility to different examples in later chapters.

Figure 5-1 Overall device physical layout

Physical layout for examples in this book: This physical layout is a base for all examples in the coming chapters. Some examples use it as shown in Figure 5-1 on page 118, and others might require changes including re-cabling. See the details in the particular chapters and examples for those changes.

5.2 Initial setup for the J08E, J16E, and J48E switches

Before configuring an IBM J08E, IBM J16E, or IBM J48E switch, the switch must be physically mounted and connected to the appropriate electrical outlets. The amount of planning and preparation that is required for the installation depends on the switch model that is being installed. Switch models have specific installation requirements, which are covered in Chapter 3, “Initial hardware planning of IBM j-type Ethernet switches and routers” on page 67.

After the switch is installed and turned on, you must set the initial configuration parameters. All of the j-type E series switches require the same initial setup.

When you turn on or restart the switch, the following actions occur:

1. The power-on self-test (POST) diagnostic tests run.
2. The Junos operating system loads.

By connecting to the switch console port using a terminal emulator, such as PuTTY, you can monitor the switch POST tests and Junos load as they progress. Example 5-1 shows a typical startup process.

Output of the software load process: The output of the software load process is provided as an example. The output will differ depending on the system hardware and configuration.

Example 5-1 Typical software load process

```
U-Boot 1.1.6 (Feb  6 2008 - 11:27:42)

Board: EX3200-24T 2.11
EPLD:  Version 6.0 (0x85)
DRAM:  Initializing (512 MB)
FLASH: 8 MB
USB:   scanning bus for devices... 2 USB Device(s) found
       scanning bus for storage devices... 1 Storage Device(s) found

Consoles: U-Boot console
Found compatible API, ver. 7

FreeBSD/PowerPC U-Boot bootstrap loader, Revision 2.1
(marcelm@apg-bbuild01.juniper.net, Wed Feb  6 11:23:55 PST 2008)
Memory: 512MB
Loading /boot/defaults/loader.conf
/kernel data=0x7f7a6c+0x877ac syms=[0x4+0x6ea20+0x4+0x9b39c]


Hit [Enter] to boot immediately, or space bar for command prompt.
Booting [/kernel]...
Kernel entry at 0xa00000e0 ...
```

```
GDB: no debug ports present
KDB: debugger backends: ddb
KDB: current backend: ddb
platform_early_bootinit: EX Series Early Boot Initialization
Copyright (c) 1996-2010, Juniper Networks, Inc.
All rights reserved.
Copyright (c) 1992-2006 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
        The Regents of the University of California. All rights reserved.
JUNOS 10.0R3.10 #0: 2010-04-16 06:31:54 UTC

builder@ormonth.juniper.net:/volume/build/junos/10.0/release/10.0R3.10/obj-powerpc
/bsd/sys/compile/JUNIPER-EX
Timecounter "decrementer" frequency 37500000 Hz quality 0
cpu0: Freescale e500v2 core revision 2.2
cpu0: HID0 80004080<EMCP,TBEN,EN_MAS7_UPDATE>
real memory  = 513802240 (490 MB)
avail memory = 502706176 (479 MB)
ETHERNET SOCKET BRIDGE initialising
Initializing EXSERIES platform properties ...
nexus0: <PPC e500 Nexus device>
ocpbus0: <on-chip peripheral bus> on nexus0
openpic0: <OpenPIC in on-chip peripheral bus> iomem 0xfef40000-0xfef600b3 on
ocpbus0
uart0: <16550 or compatible> iomem 0xfef04500-0xfef0450f irq 58 on ocpbus0
uart0: console (9600,n,8,1)
uart1: <16550 or compatible> iomem 0xfef04600-0xfef0460f irq 58 on ocpbus0
lbc0: <Freescale 8533 Local Bus Controller> iomem
0xfef05000-0xfef05fff,0xff000000-0xffffffff irq 22 on ocpbus0
cfi0: <AMD/Fujitsu - 8MB> iomem 0xff800000-0xffffffff on lbc0
syspld0 iomem 0xff000000-0xff00ffff on lbc0
tsec0: <eTSEC ethernet controller> iomem 0xfef24000-0xfef24fff irq 45,46,50 on
ocpbus0
tsec0: hardware MAC address 00:1f:12:3b:88:7f
miibus0: <MII bus> on tsec0
e1000phy0: <Marvell 88E1112 Gigabit PHY> on miibus0
e1000phy0:  10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, 1000baseTX-FDX, auto
pcib0: <Freescale MPC8544 PCI host controller> iomem
0xfef08000-0xfef08fff,0xf0000000-0xf3ffffff on ocpbus0
pci0: <PCI bus> on pcib0
pci0: <serial bus, USB> at device 18.0 (no driver attached)
ehci0: <Philips ISP156x USB 2.0 controller> mem 0xf0001000-0xf00010ff irq 22 at
device 18.2 on pci0
usb0: EHCI version 1.0
usb0: <Philips ISP156x USB 2.0 controller> on ehci0
usb0: USB revision 2.0
uhub0: Philips EHCI root hub, class 9/0, rev 2.00/1.00, addr 1
uhub0: 2 ports with 2 removable, self powered
umass0: STMicroelectronics ST72682  High Speed Mode, rev 2.00/2.10, addr 2
pcib1: <Freescale MPC8544 PCI Express host controller> iomem
0xfef0a000-0xfef0afff,0xe0000000-0xe3ffffff,0xec000000-0xec0fffff irq 42 on
ocpbus0
pci1: <PCI bus> on pcib1
pcib2: <PCI-PCI bridge> at device 0.0 on pci1
pci2: <PCI bus> on pcib2
```

```
mpfe0: <Juniper EX-series Packet Forwarding Engine> mem
0xa4000000-0xa40fffff,0xa0000000-0xa3ffffff irq 18 at device 0.0 on pci2
i2c0: <MPC85XX OnChip i2c controller> iomem 0xfef03000-0xfef03014 irq 59 on
ocpbus0
i2c1: <MPC85XX OnChip i2c controller> iomem 0xfef03100-0xfef03114 irq 59 on
ocpbus0
idma0: <mp85xxx DMA Controller> iomem 0xfef21000-0xfef21300 irq 36 on ocpbus0
Initializing product: 36 ..
bme0:Virtual BME driver initializing
Timecounters tick every 1.000 msec
Loading the NETPFE picpeer module
IPsec: Initialized Security Association Processing.
da0 at umass-sim0 bus 0 target 0 lun 0
da0: <ST ST72682 2.10> Removable Direct Access SCSI-2 device
da0: 40.000MB/s transfers
da0: 1000MB (2048000 512 byte sectors: 64H 32S/T 1000C)
if_pfe_open: listener socket opened, listening on ...
Trying to mount root from ufs:/dev/da0s1a
Attaching /packages/jbase via /dev/mdctl...
Mounted jbase package on /dev/md0...
Verified manifest signed by PackageProduction_10_0_0
Verified jboot signed by PackageProduction_10_0_0
Verified jbase-10.0R3.10 signed by PackageProduction_10_0_0
Mounted jcrypto-ex package on /dev/md1...
Verified manifest signed by PackageProduction_10_0_0
Verified jcrypto-ex-10.0R3.10 signed by PackageProduction_10_0_0
Mounted jdocs-ex package on /dev/md2...
Verified manifest signed by PackageProduction_10_0_0
Verified jdocs-ex-10.0R3.10 signed by PackageProduction_10_0_0
Mounted jkernel-ex package on /dev/md3...
Verified manifest signed by PackageProduction_10_0_0
Verified jkernel-ex-10.0R3.10 signed by PackageProduction_10_0_0
Mounted jpfe-ex42x package on /dev/md4...
Verified manifest signed by PackageProduction_10_0_0
Verified jpfe-ex42x-10.0R3.10 signed by PackageProduction_10_0_0
Mounted jroute-ex package on /dev/md5...
Verified manifest signed by PackageProduction_10_0_0
Verified jroute-ex-10.0R3.10 signed by PackageProduction_10_0_0
Mounted jswitch-ex package on /dev/md6...
Verified manifest signed by PackageProduction_10_0_0
Verified jswitch-ex-10.0R3.10 signed by PackageProduction_10_0_0
Mounted jweb-ex package on /dev/md7...
Verified manifest signed by PackageProduction_10_0_0
Verified jweb-ex-10.0R3.10 signed by PackageProduction_10_0_0
Executing /packages/mnt/jweb-ex-10.0R3.10/mount.post..
Automatic reboot in progress...
FILE SYSTEM CLEAN; SKIPPING CHECKS
clean, 39919 free (39 frags, 4985 blocks, 0.0% fragmentation)
FILE SYSTEM CLEAN; SKIPPING CHECKS
clean, 28284 free (44 frags, 3530 blocks, 0.2% fragmentation)
FILE SYSTEM CLEAN; SKIPPING CHECKS
clean, 62480 free (104 frags, 7797 blocks, 0.2% fragmentation)
FILE SYSTEM CLEAN; SKIPPING CHECKS
clean, 160932 free (28 frags, 20113 blocks, 0.0% fragmentation)
FILE SYSTEM CLEAN; SKIPPING CHECKS
```

```
clean, 160932 free (28 frags, 20113 blocks, 0.0% fragmentation)
Creating initial configuration...mgd: commit complete
Setting initial options:  debugger_on_panic=NO debugger_on_break=NO.
Starting optional daemons: .
Doing initial network setup:
.
Initial interface configuration:
additional daemons:.
Additional routing options:kern.module_path: /boot//kernel;/boot/modules ->
/boot//kernel;/Loading the NETPFE ethernet module
boot/modules;/modules
 grat_arpLoading the EX-series platform NETPFE module
_on_ifup=YES: net.link.ether.inet.grat_arp_on_ifup: 1 -> 1
 grat_arp_delay=${grat_arp_delay}: net.link.ether.inet.grat_arp_delay: 0 -> 0
kld netpfe drv: ifpfed_ethkld platform: ex_ifpfe if_vcpkern.module_path:
/boot//kernel;/boot/modules;/modules ->
/boot//kernel;/boot/modules;/modules;/modules/peertype
kld peertype: peertype_hcm peertype_pfem peertype_sfi peertype_slavere ipsec kld.
Doing additional network setup:.
Starting final network daemons:.
setting ldconfig path: /usr/lib /opt/lib
starting standard daemons: cron.
Local package initialization:.
kern.securelevel: -1 -> 1
starting local daemons:.
Fri Apr 16 09:19:12 UTC 2010

Amnesiac (ttyu0)
```

Initially, you can configure a j-series switch in one of two ways:

- Through the console using the command-line interface (CLI)
- By using the J-Web interface

These topics are described in the following sections.

5.2.1 Configuring the switch from the console

To configure the J Series switch from the console, follow these steps:

1. Insert the serial cable that is provided with the switch into the console port.
   - On the IBM J48E switch, the console port is on the rear panel of the switch.
   - On the IBM J08E switch, the console port is on the Switch Fabric and Routing Engine (SRE) module in slot SRE0.
   - On the IBM J16E switch, the console port is on the Routing Engine module in slot RE0.
2. Connect the other end of the serial cable to an RS-232 serial port on the workstation. If you do not have a male DB-9 serial port connector on your workstation, you can use a USB serial adapter.
3. Verify that the switch is on and initialization has completed.

4. Open a terminal emulator application (such as HyperTerminal or `putty.exe` on a Windows workstation or TERM in a UNIX environment), and configure it as follows:

   a. In a Microsoft Windows environment, adjust the following parameters and values if necessary:

      - Bits per second: `9600`
      - Data bits: `8`
      - Parity: **None**
      - Stop bits: `1`
      - Flow control: **None**

      Figure 5-2 shows the PuTTY serial connection configuration options

Figure 5-2 PuTTy configuration

   b. In a UNIX environment, enter the following string at the prompt:

      ```
      tip /dev/ttyb -9600
      ```

5. From the terminal emulator application, log on to the switch through the serial connection. The default administrative logon is `root`, and there is no password.

6. At the Junos shell prompt `root%`, type `ezsetup`.

> **Running the ezsetup script:** To run the **`ezsetup`** script, the switch must have the factory default configuration. If you have configured anything on the switch and want to run the **`ezsetup`** script, revert to the factory default configuration.

7. Optional: Enter the host name.

8. Enter the root password that you plan to use for this device. You are prompted to re-enter the root password.

9. Enter `yes` to enable such services as Telnet and SSH. By default, Telnet is not enabled, and SSH is enabled.

Telnet: When Telnet is enabled, you are unable to log in to an e-series switch through Telnet using root credentials. Root login is allowed only for SSH access. See “Creating a user account” on page 132 for details about creating an account for Telnet access.

10. In the Management Options panel, select the management scenario, either in-band or out-of-band.

J08E and J16E switches: On J08E and J16E switches, only the out-of-band management option is available.

11. Configure in-band management using either of the following options:
    - Use the default VLAN.
    - Create a VLAN.

      If you select this option, you are prompted to specify the VLAN name, VLAN ID, management IP address, and default gateway. Select the ports that must be part of this VLAN.
12. Configure out-of-band management. Specify the IP address and gateway of the management interface. Use this IP address to connect to the switch.
13. Optional: Specify the SNMP Read Community, Location, and Contact to configure SNMP parameters.
14. Optional: Specify the system date and time. Select the time zone from the list.
15. When you see the configured parameters, enter `yes` to commit the configuration.

The configuration is committed as the active configuration for the switch. You can now log in with the CLI or the J-Web interface to continue configuring the switch. If you use the J-Web interface to continue configuring the switch, the web session is redirected to the new management IP address. If the connection cannot be made, the J-Web interface shows instructions for starting a J-Web session.

5.2.2 Configuring the switch through the J-Web interface

To connect and configure an IBM j-type e-series Ethernet switch using the J-Web interface, follow these steps:

1. Connect the Ethernet cable from the Ethernet port on the PC to the switch:
   - On an IBM J48E switch, connect the cable to port 0 (ge-0/0/0) on the front panel of the switch.
   - On an IBM J08E or IBM J16E switch, connect the cable to the port labeled *MGMT* on the SRE module in slot SRE0 in an IBM J08E switch or on the Routing Engine module in slot RE0 in an IBM J16E switch.

   These ports are configured as a Dynamic Host Configuration Protocol (DHCP) server with the default IP address of 192.168.1.1. The switch can assign an IP address to the management PC in the IP address range 192.168.1.2 through 192.168.1.253.

Switch transition: The switch will transition to initial setup mode only when the switch is in the factory default configuration.

2. Change the switch to the initial setup mode:

   On an IBM J48E, J08E, or J16E switch, use the Menu and Enter buttons to the right of the LCD panel (see Figure 5-12 on page 134):

   a. Press the Menu button until you see MAINTENANCE MENU. Then press the Enter button.

   b. Press Menu until you see ENTER EZSetup. Then press Enter.

   c. Press Enter to confirm setup and continue with EZSetup.

   If you have configured a static IP address on your PC, you are unable to connect to the switch. To obtain an IP address dynamically, you must enable a DHCP client on the management PC that you connect to the switch.

   > **Configuration time limit:** You must complete the initial configuration using the J-Web interface within 10 minutes. The switch exits EZSetup mode after 10 minutes and reverts to the default factory configuration. Then the PC loses connectivity to the switch.
   >
   > On the IBM J48E, J08E, or J16E switch, the LCD shows a count-down timer after you connect the switch to the management PC.

3. From the PC, open a web browser, type `http://192.168.1.1` in the address field, and press Enter.

4. In the J-Web login window, type `root` as the user name, leave the password field blank, and click **Log In** (see Figure 5-3).

Figure 5-3 J-Web login window

5. In the Introduction panel (Figure 5-4), click **Next**.

Figure 5-4 EZSetup Introduction panel

6. In the Basic Settings panel (Figure 5-5), modify the host name, root password, and date and time settings:
   a. Optional: Enter the host name.
   b. Enter a password and then enter it again.
   c. Specify the time zone.
   d. Optional: Synchronize the date and time settings of the switch with the management PC or set them manually by selecting the appropriate option button.
   e. Click **Next**.

Figure 5-5 EZSetup basic settings

7. In the Management Options panel (Figure 5-6), select one of the following management scenarios and click **Next**:
   - In-band Management, use VLAN 'default' for management.

     Select this option to configure all data interfaces as members of the default VLAN.
   - In-band Management, create new VLAN for management.

     Select this option to create a management VLAN.
   - Out-of-band Management, configure management port.

     Select this option to configure only the management interface.

Figure 5-6 EZSetup management options

8. In the Management Address panel (Figure 5-7), specify the management IP address, subnet mask, and the default gateway for the default VLAN. Click **Next**.

Figure 5-7 Management address settings

9. In the Manage Access panel (Figure 5-8), select options to enable Telnet, SSH, and SNMP services. For SNMP, you can configure the read community, location, and contact. In this example, we selected the **Telnet** option. Click **Next**.

Figure 5-8 Management access options

10. Review the Summary panel (Figure 5-9) that shows the configured settings. Then click **Finish**.

Figure 5-9 EZSetup summary panel

The configuration is committed as the active switch configuration. You can now log in with the CLI or the J-Web interface to continue configuring the switch. If you use the J-Web interface to continue configuring the switch, the web session is redirected to the new management IP address. If the connection cannot be made, the J-Web interface shows instructions for starting a J-Web session.

Loss of connectivity: After the configuration takes effect, you might lose connectivity between the PC and the switch. To renew the connection, release and renew the IP address by entering the appropriate commands on the management PC or by removing and re-inserting the Ethernet cable.

Configuring the system identity for IBM j-type e-series Ethernet switches: J-Web procedure

To configure identification details for the switch, follow these steps:

1. From the J-Web home page, select **Configure**. Then in the left navigation area, select **System Properties** → **System Identity**. The System Identity panel shows the configuration details (Figure 5-10).
2. To modify the configuration, click **Edit**.

Figure 5-10 Setting the system identity properties

3. In the Edit System Identity window (Figure 5-11), enter the information as shown in Table 5-1 on page 132.

Figure 5-11 Edit System Identity window

Table 5-1 System identity configuration options

Field	Function	Action
Host Name	Defines the host name of the switching platform.	Type the host name.
Domain Name	Defines the network or subnetwork to which the machine belongs.	Type the domain name.
Root Password	Sets the root password that user `root` can use to log in to the switching platform.	Type a plain-text password. The system encrypts the password. **Note**: After a root password is defined, it is required when you log in to the J-Web user interface or the CLI.
Confirm Root Password	Verifies that the root password has been typed correctly.	Retype the password.
DNS Name Servers	Specifies the domains to be searched.	To add an IP address, click **Add**. To edit an IP address, click **Edit**. To delete an IP address, click **Delete**.
Domain Search	Specifies the domains to be searched.	To add a domain, click **Add.** To edit a domain click **Edit**. To delete a domain, click **Delete**.

Creating a user account

To allow login by using Telnet, a user account must be created and a class set. To log on to a device running Junos software using a root account and configure a new user account, follow these steps:

1. Log in as root and enter configuration mode:

   ```
   root@host> configure
   [edit]
   root@host#
   ```

   The prompt in brackets (`[edit]`), also known as a *banner*, shows that you are in configuration edit mode, at the top of the hierarchy.

2. Change to the `[edit system login]` section of the configuration:

   ```
   [edit]
   root@host# edit system login
   [edit system login]
   root@host#
   ```

   The prompt in brackets changes to `[edit system login]` to show that you are at a new level in the hierarchy.

3. Add a new user account:

   ```
   [edit system login]
   root@host# edit user username
   ```

   This example adds an account *`username`*, but you can use any account name.

4. Configure a full name for the account. If the name includes spaces, enclose the entire name in quotation marks (" "):

   ```
   [edit system login user username]
   root@host# set full-name "full name"
   ```

5. Configure an account class. The account class sets the user access privileges for the account:

```
[edit system login user username]
root@host# set class super-user
```

6. Configure an authentication method and password for the account:

```
[edit system login user username
root@host# set authentication plain-text-password
New password:password
Retype new password:password
```

When the new password prompt is displayed, enter a clear-text password that the system will encrypt, and then confirm the new password.

7. Commit the configuration:

```
[edit system login user username]
root@host# commit
commit complete
```

Configuration changes are not activated until you commit the configuration. If the commit is successful, a commit complete message is displayed.

8. Return to the top level of the configuration, and then exit:

```
[edit system login user username]
root@host# top
[edit]
root@host# exit
Exiting configuration mode
```

9. Log out of the device:

```
root@host> exit
% logout Connection closed.
```

10. To test your changes, log back in with the user account and password that you just configured:

```
login: username
Password: password
--- JUNOS 9.5R1.8 built 2009-04-13 19:25:06 UTC
username@host>
```

When you log in, you see the new user name at the command prompt.

5.3 LCD panel and status LEDs

The IBM J08E, IBM J16E, and IBM J48E switches are provided with a front panel LCD display and status LEDs.

5.3.1 LCD panel

The LCD panel on the top front of the IBM J08E, IBM J16E, and IBM J48E switches shows two lines of text with a maximum of 16 characters in each line. The LCD panel shows various information about the switch. It also provides menu options to perform basic operations such as initial configuration and switch reboot.

There are two navigation buttons, Menu and Enter, to the right of the LCD panel, as shown in Figure 5-12.

Figure 5-12 IBM J48E front panel

The LCD panel has a backlight. If the LCD panel is idle for 60 seconds, the backlight turns off. You can turn on the backlight by pressing the Menu or Enter button once. After turning on the backlight, you can toggle between the LCD menus by pressing the Menu button and navigate through the menu options by pressing the Enter button.

Chassis viewer: The chassis viewer in the J-Web interface also shows the LCD panel. From the J-Web interface, you can view real-time status information in the LCD panel.

LCD panel modes

The LCD operates in four modes: boot, idle, status, and maintenance.

The LCD operates in boot mode during switch reboot. The boot mode shows the key milestones in the switch boot process. The boot mode does not have any menu options. After the boot process is complete, the LCD automatically reverts to the Idle menu.

In an IBM J48E switch that is not a member of a Virtual Chassis, the first line of the LCD panel shows the slot number, the role of the switch, and the host name. For a stand-alone IBM J48E switch, the slot number is always *00,* and the role is always *RE* (for master).

In a J48E switch that is a member of a Virtual Chassis, the first line of the LCD panel shows the following information:

- The slot number (the member ID for the Virtual Chassis member)
- Role of the switch in a Virtual Chassis (*RE* for master, *BK* for backup, and *LC* for linecard member)
- Host name

In the idle mode, line two of the Idle menu shows the Status LED modes of the network port and the total number of alarms in the system. The number of alarms is updated every second.

With the status mode, you can get status information for the following items:

- Switch fabric in SRE modules in IBM J08E switches
- Routing Engine and switch fabric in switch fabric modules in IBM J16E switches
- Power supplies
- Fan tray or trays and chassis temperature

- Junos software version installed
- Virtual Chassis port (VCP) status (for an IBM J48E switch that is a member of a Virtual Chassis)

With the maintenance mode, you can cycle through options for configuring and troubleshooting the switch:

- System halt
- Reboot system
- Load rescue configuration
- Revert to factory default configuration
- EZSetup

LCD panel menus for IBM J08E and IBM J16E switches

The LCD has three menus: Idle, Status, and Maintenance. In each of these menus, line one of the LCD panel shows the host name of the switch. You can toggle between the LCD menus by pressing the Menu button and navigate through the menu options by pressing the Enter button.

Table 5-2 describes the LCD menu options.

Table 5-2 IBM J08E and J16E switch LCD panel menus

Menu	Description

Idle	In the Idle menu:
	- Press Enter to cycle through the Status LED modes, which are port status indicators:
	  - ADM (enabled/disabled)
	  - SPD (speed)
	  - DPX (duplex)
	- Press Menu to exit the Idle menu and go to the Status menu.

Status	The Status menu has the following options:
	- Switch fabric status. Choose one of the following options:
	  - Press Enter to view the status of the switch fabric in the SRE modules (SRE0 and SRE1) in IBM J08E switches and the SF modules (SF) in IBM J16E switches: OK, Fld (failed), ABS (absent)
	  - Press Menu to go to the next option in the Status menu.
	- Power supply status (1). Choose one of the following options:
	  - Press Enter to view the status of power supplies 0 and 1: OK, Fld, ABS.
	  - Press Menu to go to the next option in the Status menu.
	- Power supply status (2). Choose one of the following options:
	  - Press Enter to view the status of power supplies 2, 3, 4, and 5: OK, Fld, ABS.
	  - Press Menu to go to the next option in the Status menu.
	- Environment status. Choose one of the following options:
	  - Press Enter to view the status of the fan tray or trays and the chassis temperature:
	  - Fan tray status: OK, Fld, ABS
	  - Chassis temperature status: OK, High, Shutdown
	  - Press Menu to go to the next option in the Status menu.
	- Junos version status. Choose one of the following options:
	  - Press Enter to see the version of Junos software loaded on the switch.
	  - Press Menu to go to the next option in the Status menu.
	- EXIT STAT MENU? Choose one of the following options:
	  - Press Enter to exit the Status menu.
	  - Press Menu to return to the Switch fabric status option.

Menu	Description
Maintenance	The Maintenance menu has the following options: ► SYSTEM HALT?. Choose one of the following options: ► Press Enter to halt the master SRE module in an IBM J08E switch or to halt the master Routing Engine module in an IBM J16E switch. Press Enter again to confirm the halt. In a base configuration switch, the master SRE or Routing Engine module is gracefully halted but not powered off. Press Enter on your management device or power cycle the switch to start it again. In a redundant configuration, the backup SRE or Routing Engine module takes over mastership when the master SRE or Routing Engine module is halted. To completely halt the switch, use the **`request system halt other-routing-engine`** CLI command to halt the backup SRE or Routing Engine module before halting the master SRE or Routing Engine module. Press Enter on your management device or power cycle the switch to start it again. ► Press Menu to go to the next option in the Maintenance menu. ► SYSTEM REBOOT? Choose one of the following options: ► Press Enter to reboot the master SRE or Routing Engine module. Press Enter again to confirm the reboot. ► Press Menu to go to the next option in the Maintenance menu. ► LOAD RESCUE? Choose one of the following options: ► Press Enter to roll back the switch to the previous valid configuration. Press Enter again to confirm the rollback. ► Press Menu to go to the next option in the Maintenance menu. ► FACTORY DEFAULT? Choose one of the following options: ► Press Enter to restore the switch to the factory default configuration. Press Enter again to confirm the restoration. The LCD flashes a success or failure message and returns to the Idle menu. ► Press Menu to go to the next option in the Maintenance menu. ► ENTER EZSETUP? Choose one of the following options: ► Press Enter to launch EZSetup. Press Enter again to confirm the launch. EZSetup configures DHCP and enables the J-Web user interface on the switch. The LCD flashes a success or failure message for approximately 10 seconds and returns to the Idle menu. ► Press Menu to go to the next option in the Maintenance menu. **EZSetup**: You can use the EZSetup option only if the switch is in the factory default configuration. ► EXIT MAINT MENU? Choose one of the following options: ► Press Enter to exit the Maintenance menu. ► Press Menu to return to the SYSTEM HALT option. You can disable the Maintenance menu in the LCD panel by using the **`set chassis lcd maintenance-menu disable`** command in the CLI.

LCD panel menus for the IBM J48E switch

The LCD has three menus: Idle, Status, and Maintenance. In each of these menus, line one of the LCD panel shows the host name of the switch. You can toggle between the LCD menus by pressing the Menu button and select the menu options by pressing the Enter button.

Table 5-3 describes LCD menu options.

Table 5-3 IBM J48E switch LCD panel menus

Menu	Description
Idle	In the Idle menu: ► Press Enter to cycle through the Status LED modes, which are port status indicators: ► ADM (enabled/disabled) ► SPD (speed) ► Power over Ethernet (PoE) ► DPX (duplex) ► EXIT IDLE MENU? Select this option to exit the Idle menu.
Status	The Status menu has the following options: ► Show VCP Status: Shows the Virtual Chassis port (VCP) status: Up, Down, Disabled. This menu option is available only for a J48E switch that is a member of a Virtual Chassis configuration. ► Show PSU Status: Shows the status of the power supply: OK, Failed, Absent. ► Show Environment Status: Shows the status of the fan and temperature: ► Fan status: OK, Failed, Absent ► Temp status: OK, High, Shutdown ► Show Junos Version Status: Shows the version of Junos software loaded on the switch. ► EXIT STAT MENU? Select this option to exit the Status menu.
Maintenance	The Maintenance menu has the following options to configure and troubleshoot the switch: ► SYSTEM HALT? Select this option by using the Enter button to halt the switch. Press the Enter button again to confirm the halt. ► Press the Menu button to go to the next option in the Maintenance menu. ► SYSTEM REBOOT? Select this option by using the Enter button to reboot the switch. Press the Enter button again to confirm the reboot. ► Press the Menu button to go to the next option in the Maintenance menu. ► LOAD RESCUE? Select this option by using the Enter button to roll back the switch to the rescue configuration. Press the Enter button again to confirm the rollback. ► Press the Menu button to go to the next option in the Maintenance menu. ► FACTORY DEFAULT? Select this option by using the Enter button to restore the switch to the factory default configuration. Press the Enter button again to confirm the restoration. ► Press the Menu button to go to the next option in the Maintenance menu. ► ENTER EZSETUP? Select this option by using the Enter button to launch EZSetup. Press the Enter button again to confirm the launch. ► Press the Menu button to go to the next option in the Maintenance menu. **Note**: You can use this option only if the switch is in the factory default configuration. ► EXIT MAINT MENU? Select this option to exit the Maintenance menu. You can disable the Maintenance menu in the LCD panel by using the **`set chassis lcd maintenance-menu disable`** command in the CLI.

5.3.2 Chassis status LEDs

The front panel of the IBM J08E, IBM J16E, and IBM J48E switches have three status LEDs on the right side of the LCD panel.

Status LEDs for IBM J08E, and IBM J16E switches

Table 5-4 on page 138 describes the chassis status LEDs in an IBM J08E or J16E switch, their colors and states, and the status they indicate. You can view the colors of the three LEDs remotely through the CLI by issuing the **`show chassis lcd`** operational mode command.

Table 5-4 IBM J08E and J16E Status LEDs

LED label (description)	Color	State and description
ALM (Alarm)	Unlit	No alarm.
	Red	Major alarm.
	Yellow	Minor alarm.
SYS (System)	Unlit	Switch is powered off.
	Yellow	One or more component failures are generating one or more alarms.
	Green	Switch is operating normally.
MST (Master)	Unlit	Switch is powered off.
	Green	Master Routing Engine is operational.

A major alarm (red) indicates a critical error condition that requires immediate action. A minor alarm (yellow) indicates a noncritical condition that requires monitoring or maintenance. A minor alarm that is not checked might cause interruption in service or performance degradation.

All three LEDs can be lit simultaneously.

Status LEDs for IBM J48E switch

Table 5-5 describes the chassis status LEDs in an IBM J48E switch, their colors and states, and the status they indicate. You can view the colors of the three LEDs remotely through the CLI by issuing the **`show chassis lcd`** operational mode command.

Table 5-5 IBM J48E status LEDs

LED label (description)	Color	State and description
ALM (Alarm)	Unlit	No alarm.
	Red	Major alarm.
	Amber	Minor alarm.
SYS (System)	Green	On steadily: The Junos software for EX Series switches has been loaded on the switch.
		Blinking: The switch is booting.
MST (Master)	Green	On steadily: The switch is the master in the Virtual Chassis configuration.
		Blinking: The switch is the backup in the Virtual Chassis configuration.
		Off: The switch is a linecard member in the Virtual Chassis configuration.

A major alarm (red) indicates a critical error condition that requires immediate action.

A minor alarm (amber) indicates a noncritical condition that requires monitoring or maintenance. A minor alarm that is not checked might cause interruption in service or performance degradation.

Amber alarm: The amber glow of the Alarm LED that indicates a minor alarm closely resembles the red glow that indicates a major alarm.

All three LEDs can be lit simultaneously.

5.4 Initial setup for the J02M, J06M, and J11M routers

The IBM J02M, J06M, and J11M Multiservice Edge Routers are shipped with the Junos software preinstalled and ready to be configured when the router is powered on. Three copies of the software are provided:

- One on a compact flash card in the Routing Engine
- One on a rotating hard disk in the Routing Engine
- One on a USB flash drive that can be inserted into the slot in the Routing Engine faceplate

When the router boots, it first attempts to start the image on the USB flash drive. If a USB flash drive is not inserted into the Routing Engine or the attempt otherwise fails, the router next tries the compact flash card (if installed) and finally the hard disk.

5.4.1 Initial configuration

You configure the router by issuing Junos CLI commands in either of the following ways:

- On a console device attached to the console port on the Routing Engine
- Over a Telnet connection to a network connected to the Ethernet port on the Routing Engine.

Before you configure the router, you must gather the following information:

- The name that the router will use in the network
- The domain name that the router will use
- The IP address and prefix length information for the Ethernet interface
- The IP address of a default router
- The IP address of a DNS server
- The password for the root user

This procedure connects the router to the network but does not enable it to forward traffic. For complete information about enabling the router to forward traffic, including examples, see Chapter 10, “Layer 3: Routing configuration” on page 273.

Configuring the router from the console

To configure the router from the console, follow these steps:

1. Insert the serial cable that is provided with the router into the console port on the Routing Engine.
2. Connect the other end of the serial cable to an RS-232 serial port on the workstation. If you do not have a male DB-9 serial port connector on your workstation, you can use a USB serial adapter.
3. Verify that the switch is on and initialization has completed.

4. Open a terminal emulator application (such as HyperTerminal or `putty.exe` program on a Windows workstation or TERM in a UNIX environment), and configure it as follows:

   a. In Microsoft Windows environment, adjust the following parameters and values if necessary:

      - Bits per second: `9600`
      - Data bits: `8`
      - Parity: **None**
      - Stop bits: `1`
      - Flow control: **None**

      Figure 5-13 shows the PuTTY serial connection configuration options.

Figure 5-13 Serial port settings for Router Console

   b. In a UNIX environment, enter the following string at the prompt:

      ```
      tip /dev/ttyb -9600
      ```

5. From the terminal emulator application, log on to the router through the serial connection. The default administrative logon is **root** and there is no password.

   ```
   login: root
   Password:
   ```

6. Start the CLI:

   ```
   root@% cli
   root>
   ```

7. Enter configuration mode:

   ```
   root> configure
   Entering configuration mode

   [edit]
   root#
   ```

8. Configure the name of the router. If the name includes spaces, enclose the name in quotation marks (" "):

```
[edit]
root# set system host-name hostname
```

9. Create a management console user account:

```
[edit]
root# set system login user username authentication plain-text-password
New password:password
Retype new password:password
```

10. Set the users full name. If the name includes spaces, enclose the entire name in quotation marks (" "):

```
[edit]
root# set system login user username full-name "Full Name"
```

11. Set the user account class to super-user:

```
[edit]
root# set system login user username class super-user
```

12. Configure the domain name of the router:

```
[edit]
root# set system domain-name domain-name
```

13. Configure the IP address and prefix length for the Ethernet interface of the router.

```
[edit]
root# set interfaces fxp0 unit 0 family inet address address/subnet-mask
```

14. Configure the IP address of a backup router, which is used only while the routing protocol is not running:

```
[edit]
root# set system backup-router address
```

15. Configure the IP address of a DNS server:

```
[edit]
root# set system name-server address
```

16. Set the root authentication password by entering one of the following options:
    - A clear-text password:

      ```
      [edit]
      root# set system root-authentication plain-text-password
      New password:password
      Retype new password:password
      ```

    - An encrypted password

      ```
      [edit]
      root# set system root-authentication encrypted-password encrypted-password
      ```

    - An SSH public key string (DSA or RSA)

      ```
      [edit]
      root# set system root-authentication ssh-dsa public-key string

      [edit]
      root# set system root-authentication ssh-rsa public-key string
      ```

17. Optional: Configure the static routes to remote subnets with access to the management port. Access to the management port is limited to the local subnet. To access the management port from a remote subnet, add a static route to that subnet within the routing table.

```
[edit]
root# set routing-options static route remote-subnet next-hop IP Address retain
no-readvertise
```

18. Configure the Telnet service:

```
[edit]
root# set system services telnet
```

19. Optional: View the configuration to verify that it is correct.

```
[edit]
root# show
system {
   host-name host-name;
   domain-name domain-name;
   backup-router address;
   root-authentication {
      authentication-method password;
   }
   name-server {
      address;
   }
}
interfaces {
   fxp0 {
      unit 0 {
         family inet {
            address address/prefix-length;
         }
      }
   }
}
```

20. Commit the configuration to activate it on the router:

```
[edit]
root# commit
```

21. After configure the router, exit the configuration mode:

```
edit]
root# exit
root>
```

6

User interface

This chapter provides information about the following user interfaces that you can use to configure, manage, monitor and troubleshoot IBM j-type Ethernet switches and routers:

- J-Web graphical user interface (GUI)
- Junos command-line interface (CLI)

While most of the configuration steps in this book use CLI, this chapter provides a brief introduction for the J-Web GUI. For more information about the J-Web GUI, see the *J-Web User Interface Guide* at:

http://jnpr.net/techpubs/en_US/junos10.1/information-products/topic-collections/jweb-user-guide/frameset.html

This chapter includes the following topics:

- J-Web GUI
- Junos CLI

© Copyright IBM Corp. 2011. All rights reserved.

6.1 J-Web GUI

The J-Web interface is a web-based GUI that you can access by using either HTTP or HTTP Secure (HTTPS). It provides a GUI to perform basic configuration tasks and complex settings using CLI tools from J-Web GUI.

The J-Web GUI is installed by default on IBM j-type Ethernet switches. It is not installed on IBM j-type Ethernet routers. You must install the J-Web GUI on routers when you want to use it.

6.1.1 Logging in to the J-Web GUI

Before you can access the J-Web interface, you must ensure that HTTP or HTTPS has been enabled. You must also have a valid account for logging in and for authentication as shown in Figure 6-1.

Figure 6-1 J-Web login window

Use Internet Explorer version 6.0 and later, or Firefox version 2.0 and later to access the J-Web interface.

Secure web access overview

A routing platform uses the Secure Sockets Layer (SSL) protocol to provide secure management of routing platforms through the web interface. SSL uses public-private key technology that requires a paired private key and an authentication certificate for the SSL service. SSL encrypts communication between your routing platform and the web browser with a session key negotiated by the SSL server certificate.

An SSL certificate includes identifying information such as a public key and a signature made by a certificate authority (CA). When you access the routing platform through HTTPS, an SSL handshake authenticates the server, and the client and begins a secure session. If the information does not match or the certificate has expired, you are unable to access the routing platform through HTTPS.

Without SSL encryption, communication between your routing platform and the browser is sent in the open and can be intercepted. You must enable HTTPS access on your WAN interfaces.

On j-type switches, and routers, HTTP access is enabled by default on the built-in management interfaces. By default, HTTPS access is supported on any interface with an SSL server certificate.

Generating SSL certificates

To enable secure web access, you must first generate a digital SSL certificate and then enable HTTPS access on the routing platform.

To generate an SSL certificate, follow these steps:

1. Enter the following **openssl** command in your Secure Shell (SSH) CLI:

   ```
   % openssl req –x509 –nodes –newkey rsa:1024 –keyout filename.pem -out
   filename.pem
   ```

   Replace *filename* with the name of a file in which you want the SSL certificate to be written, for example, `new.pem`.

   The **openssl** command generates a self-signed SSL certificate in the privacy-enhanced mail (PEM) format. It writes the certificate and an unencrypted 1024-bit RSA private key to the specified file.

2. When prompted, type the appropriate information in the identification form. For example, type `US` for the country name.

3. Show the contents of the file `new.pem`:

   ```
   cat new.pem
   ```

   Copy the contents of this file to install the SSL certificate.

To install the SSL certificate and enable HTTPS, follow the procedure as outlined in “Configuring secure web access” on page 146.

Configuring secure web access

Navigate to the Management Access Configuration window by selecting **Configuration** → **System Properties** → **Management Access**. Click **Edit** in the main window. Then in the Edit Management Access window (Figure 6-2), you can enable HTTP and HTTPS access on interfaces for managing services routers through the web interface. You can also install SSL certificates and enable JUNOScript over SSL with the Secure Access window.

Figure 6-2 Edit Management Access window: Services and Certificates tabs

To configure web access settings in the J-Web interface, enter the information in the Edit Management Access window as follows:

- HTTP web access
  - Enable HTTP Access: To enable HTTP access, select the **Enable HTTP** check box on the **Services** tab.
  - Enable HTTP on All Interfaces: To enable HTTP access on all interfaces, select the **Enable on all interfaces** check box on the **Services** tab.
  - To enable HTTP on select interfaces, clear the **Enable on all interfaces** check box on the **Services** tab, select the interface, and move it to the appropriate list by clicking the direction arrows:
    - To enable HTTP access on an interface, add the interface to the Selected interfaces list.
    - To disable HTTP access on an interface, add the interface to the Available interfaces list.

- HTTPS web access
  - Enable HTTPS Access: To enable HTTPS access, select the **Enable HTTPS** access check box on the **Services** tab.
  - HTTPS Certificate: This option specifies SSL certificates to be used for encryption. It is available only after you create an SSL certificate as explained in “Generating SSL certificates” on page 145. To specify the HTTPS certificate, select a certificate from the HTTPS certificate list on the **Services** tab.
  - Enable HTTPS on All Interfaces: To enable HTTPS on all interfaces, select the **Enable on all interfaces** check box on the **Services** tab.
  - To enable HTTPS on select interfaces, clear the **Enable on all interfaces** check box on the **Services** tab, select the interface, and move it to the appropriate list by clicking the direction arrows:
    - To enable HTTPS access on an interface, add the interface to the Selected interfaces list.
    - To disable HTTPS access on an interface, add the interface to the Available interfaces list.

JUNOScript over SSL: With JUNOScript over SSL, you can enable secured SSL access to the JUNOScript XML scripting API. A JUNOScript client, such as JUNOScope, is required.

After selecting the appropriate options, click **OK** to apply the configuration.

To verify that web access is enabled correctly, connect to the router by using one of the following methods:

- For HTTP access, in your web browser, type `http://URL` or `http://IP address`.
- For HTTPS access, in your web browser, type `https://URL` or `https://IP address`.

6.1.2 Dashboard tab

When you first access the J-Web interface, you see a chassis view of the device, the Dashboard tab (Figure 6-3), which is the default tab of the J-Web interface.

Figure 6-3 Dashboard tab

On the Dashboard tab, you can view the following details:

- Graphical Chassis, which shows the port status, alarms, and the LCD screen of the front view of the chassis. You can also click the **Rear View** button to see the rear view of the chassis.
- System Information box, which shows such information as system name, device model, Junos version, and other details related to the system.
- Capacity Utilization box, which shows the usage of ports, MAC table entries, and configured VLAN.
- Health Status box, which shows environmental status such as memory utilization, flash usage, temperature, processor load, and fan condition.
- Alarms box, which reports the red and yellow alarms of the switch.

6.1.3 Configure tab

The Configure tab (Figure 6-4) provides an easy-to-use interface to configure and view your switch.

Figure 6-4 Configure tab

On the left side of the window is the configuration hierarchy. When you click one of the configuration hierarchy options, the information about the hierarchy is displayed on the main portion of the window. Here you can view or edit the configuration. In the top right corner of the window, you see Add, Edit, and Delete buttons for performing the configuration.

The CLI Tools option, on the bottom of configuration hierarchy, provides a flexible text editor that functions similarly to the CLI of Junos.

6.1.4 Monitor tab

On the Monitor tab, you can view the real-time information and statistics shown in Figure 6-5.

Figure 6-5 Monitor tab

For example, on the interface hierarchy, the Monitor tab provides statistics in graphical and colorful pie charts and graphs. If you move the mouse pointer to various part of the window, you are presented with more detailed information.

6.1.5 Maintain tab

The Maintain tab provides easy file and software management functions as shown in Figure 6-6.

Figure 6-6 Maintain tab

You can perform maintenance of your Ethernet devices by using the following options:

- Files. You can download and delete log files and temporary files to keep the compact-flash device from becoming full.
- Configuration Management. With this option, you can retrieve historical configuration files and compare the differences between configurations.
- Software. This option provides easy ways to upgrade and downgrade the Junos software.
- Licenses. This option shows the installed licenses and feature summary.
- Reboot. With this option, you can schedule various methods of rebooting the switch.
- Customer Support. Use this option to view detailed information of the switch that you can use when contacting Technical Assistance Center (TAC) for technical assistance.

6.1.6 Troubleshoot tab

The Troubleshoot tab (Figure 6-7) provides tools for basic troubleshooting without openning a CLI session.

Figure 6-7 Troubleshoot tab

You can use quick and useful troubleshooting tools such as Ping, Traceroute, Packet capture, RPM, and CLI Terminal to investigate the network problem.

6.2 Junos CLI

Junos CLI is a specific command shell from Juniper Networks that runs on top of a UNIX-based operating system kernel. The CLI provides command help and command completion.

The CLI also provides various UNIX utilities, such as the following examples:

- Emacs-style keyboard sequences so that you can move around on a command line and scroll through recently executed commands
- Regular expression matching to locate and replace values and identifiers in a configuration, filter command output, or log file entries
- Store and archive router files on a UNIX-based file system
- Exit from the CLI environment and create a UNIX C shell or Bourne shell to navigate the file system
- Manage switch processes

6.2.1 CLI Modes

The CLI has two modes:

- Operational mode
- Configuration mode

Operational mode is where you can troubleshoot and monitor the software, router, and network. The right angle bracket (>) is the identifier in operational mode as shown in the following example:

```
user@J48E>
```

Configuration mode is where the actual statements for interfaces, routing protocols, and others are placed. The number sign (#) is the identifier in configuration mode as shown in the following example:

```
user@J48E#
```

6.2.2 Logging in CLI

Because Junos runs on top of UNIX environment, you must have a user name and password to log in. A root user is created by default. You log in to CLI using root for first time and then create other user accounts and assign permissions.

When you first enter the switches and routers by using Telnet, SSH, or direct console access, you see a login prompt. After entering the correct user name and password, you are placed directly into operational mode shown in Example 6-1.

Example 6-1 Logging in to CLI

```
J48E (ttyp0)

login: ibm
Password:

--- JUNOS 10.1R1.8 built 2010-02-12 17:24:20 UTC
{master:0}
ibm@J48E>
```

An exception case of being placed into operational mode is when you log in as root user. In this case, you are placed in the shell (designated by the percent sign (%)) and must start the CLI manually by using the **cli** command as shown in Example 6-2.

Example 6-2 Logging in to CLI as a root user

```
J48E (ttyp0)

login as: root
root@10.1.1.10's password:

--- JUNOS 10.1R1.8 built 2010-02-12 17:24:20 UTC

root@J48E:RE:0% cli
{master:0}
root@J48E>
```

6.2.3 CLI operational mode

Operational mode CLI commands are used to monitor and control the operation of the switches and routers. The operational mode CLI commands are hierarchically structured as shown in Figure 6-8.

Figure 6-8 Hierarchy of commands

The commands are run from the default CLI level (**user@switch>**). For example, to see the brief routes of current routing table, you can enter the **show route brief** command as shown in Example 6-3.

Example 6-3 Output of the show route brief command

```
ibm@J48E> show route brief

inet.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.1.1.0/24        *[Direct/0] 1d 00:17:14
                    > via me0.0
10.1.1.10/32       *[Local/0] 1d 22:53:26
                       Local via me0.0
192.168.1.0/30     *[Direct/0] 1d 03:50:53
                    > via ge-0/0/0.0
192.168.1.1/32     *[Local/0] 1d 03:50:53
                       Local via ge-0/0/0.0
224.0.0.2/32       *[PIM/0] 1d 22:53:28
                       MultiRecv
224.0.0.13/32      *[PIM/0] 1d 22:53:28
                       MultiRecv

inet6.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
```

```
ff02::2/128        *[PIM/0] 1d 22:53:28
                      MultiRecv
ff02::d/128        *[PIM/0] 1d 22:53:28
                      MultiRecv
```

Key operational-mode capabilities

Junos software offers the following key operational-mode capabilities:

- Entering configuration mode
- Controlling the CLI environment
- Exiting CLI
- Monitoring and troubleshooting
  - clear
  - monitor
  - ping
  - show
  - test
  - traceroute
- Connecting to other network systems
- Copying files
- Restarting software processes
- Performing system-level operations

Command completion

The command completion feature saves a lot of time and energy because it provides syntax checking as you type. You no longer must type a command on a line and press Enter to find that the command is either invalid or not supported on that version of software. In Junos, any error or ambiguity is detected early, and the router and switch present a list of possible valid completions.

Command completion is accomplished using either the Spacebar or the Tab key. To complete a command, statement, or option that you partially typed, press the Tab key or the Spacebar. If the partially typed letters uniquely identify a command, the complete command name is displayed. Otherwise, a beep indicates that you have entered an ambiguous command and the possible completions are displayed. This completion feature also applies to other strings, such as file names, interface names, user names, and configuration statements.

For example, to view the configuration of a certain Gigabit Ethernet interface, you can type the command as shown in Example 6-4.

Example 6-4 Using the Spacebar and tab key

```
user@J48E> sh<space>ow conf<space>iguration int<space>erfaces g<tab>e-0/0/0
<enter>
```

The Spacebar is used until a variable is reached and the interface name is used when the Tab key must be used. (The Spacebar completes only commands and not variables.)

In Example 6-4, the syntax checker goes word by word each time that the Spacebar or Tab key is used and minimum characters are typed to avoid ambiguity.

In a case where syntax checker notices that an error is an incomplete word, it states the ambiguity and provides the possible completions as shown in Example 6-5.

Example 6-5 Using Spacebar to list possible completions

```
ibm@J48E> show ip<space>
                 ^
'ip' is ambiguous.
Possible completions:
  ip-source-guard      Show IP source guard information
  ipsec                Show IP Security information
  ipv6                 Show IP version 6 information
```

Emacs

Junos software defaults to a VT100 terminal type. With this terminal type, users of keyboard arrow keys can have any additional configuration modification. You can use Emacs-style keystrokes so that you can move the cursor around the command line, edit the command line, or delete specific characters or words. The following Emacs keystrokes are useful:

Ctrl+B Moves the cursor left one character.

Ctrl+A Moves the cursor to the beginning of the command line.

Ctrl+F Moves the cursor right one character.

Ctrl+E Moves the cursor to the end of the command line.

Ctrl+D Deletes the character over the cursor.

Ctrl+K Deletes from the cursor to the end of the line.

Ctrl+U Deletes all the characters and negates the current command.

Ctrl+W Deletes the entire word to the left of the cursor.

Ctrl+L Redraws the current line.

Ctrl+P Scrolls backward through the previously typed commands. You also can use the Up arrow for this purpose.

Ctrl+N Scrolls forward through the previously typed commands. You also can use the Down arrow for this purpose.

Ctrl+R Search the previous CLI history for a search string.

Pipe commands

Another useful feature of Junos CLI is the use of pipe commands to control the output of any command. When a command, such as **show**, is issued, the data is placed into a buffer and is displayed when the Enter key is pressed. A pipe command can be used to alter the display buffer.

The following sections examine the most common applications of pipe commands.

The count command

Use the **count** command to count the lines in the output as shown in Example 6-6.

Example 6-6 Pipe with count command

```
ibm@J48E> show configuration | count
Count: 352 lines
```

The display command

Use the **display** command to view additional information, such as **set** commands, as shown in Example 6-7.

Example 6-7 Pipe with the display set command

```
ibm@J48E> show configuration | display set
set version 10.1R1.8
set system host-name J48E
set system root-authentication encrypted-password
"$1$U7AW/vKF$UUNnJTiq3psmzUh1mRTe/0"
set system login user ibm uid 2001
set system login user ibm class super-user
set system login user ibm authentication encrypted-password
"$1$qb66ubMo$bhUmZ/iY3aPYEPCUuaWEn."
set system services ssh
set system services telnet
set system services web-management http
set system syslog user * any emergency
set system syslog file messages any notice
set system syslog file messages authorization info
set system syslog file interactive-commands interactive-commands any
set interfaces ge-0/0/0 unit 0 family inet address 192.168.1.1/30
set interfaces ge-0/0/1 unit 0 family ethernet-switching
set interfaces ge-0/0/2 unit 0 family ethernet-switching
set interfaces ge-0/0/3 unit 0 family ethernet-switching
set interfaces ge-0/0/4 unit 0 family ethernet-switching
```

The except command

Use the **except** command to view only text that does not match a pattern as shown in Example 6-8.

Example 6-8 Pipe with the except command

```
ibm@J48E> show interfaces terse | except ge
Interface               Admin Link Proto    Local                 Remote
vcp-0                   up    down
vcp-0.32768             up    down
vcp-1                   up    down
vcp-1.32768             up    down
bme0                    up    up
bme0.32768              up    up   inet     128.0.0.1/2
                                            128.0.0.16/2
                                            128.0.0.32/2
                                      tnp    0x10
dsc                     up    up
gre                     up    up
ipip                    up    up
lo0                     up    up
lsi                     up    up
me0                     up    up
me0.0                   up    up   inet     10.1.1.10/24
mtun                    up    up
pimd                    up    up
pime                    up    up
tap                     up    up
```

```
vlan                    up    up
vlan.0                  up    up   inet
vme                     up    down
vme.0                   up    down inet
```

The find command

Use the **find** command to search for first occurrence of a pattern as shown in Example 6-9.

Example 6-9 Pipe with the find command

```
ibm@J48E> show interfaces ge-0/0/0 extensive | find traffic
  Traffic statistics:
   Input  bytes  :            427368738                    0 bps
   Output bytes  :            430971601                    0 bps
   Input  packets:               542474                    0 pps
   Output packets:               550223                    0 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3
incompletes: 0, L2 channel errors: 0,
    L2 mismatch timeouts: 0, FIFO errors: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 13, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0,
FIFO errors: 0,
    HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Egress queues: 8 supported, 4 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort                    0               543076                    0
    1 assured-forw                   0                    0                    0
    5 expedited-fo                   0                    0                    0
    7 network-cont                   0                 7147                    0
  Active alarms  : None
  Active defects : None
```

The match command

Use the **match** command to see only text that matches a pattern as shown in Example 6-10.

Example 6-10 Pipe with the match command

```
ibm@J48E> show log messages | match "feb 13"
Feb 13 00:08:45  J48E chassisd[802]: CHASSISD_SNMP_TRAP6: SNMP trap generated:
Power Supply failed (jnxContentsContainerIndex 2, jnxContentsL1Index 1,
jnxContentsL2Index 2, jnxContentsL3Index 0, jnxContentsDescr Power Supply 1,
jnxOperatingState/Temp 6)
Feb 13 01:08:45  J48E chassisd[802]: CHASSISD_SNMP_TRAP6: SNMP trap generated:
Power Supply failed (jnxContentsContainerIndex 2, jnxContentsL1Index 1,
jnxContentsL2Index 2, jnxContentsL3Index 0, jnxContentsDescr Power Supply 1,
jnxOperatingState/Temp 6)
Feb 13 02:08:46  J48E chassisd[802]: CHASSISD_SNMP_TRAP6: SNMP trap generated:
Power Supply failed (jnxContentsContainerIndex 2, jnxContentsL1Index 1,
```

```
jnxContentsL2Index 2, jnxContentsL3Index 0, jnxContentsDescr Power Supply 1,
jnxOperatingState/Temp 6)
```

Context-sensitive help

The Junos CLI provides context-sensitive help at any point in a command line. The help indicates which options are acceptable at the current point in the command and provides a brief description about each command or command option.

To receive help at any time while in the Junos CLI, type a question mark (?) as shown in Example 6-11. You do not need to press Enter.

Example 6-11 Help by typing a question mark

```
ibm@J48E> ?
Possible completions:
  clear                Clear information in the system
  configure            Manipulate software configuration information
  file                 Perform file operations
  help                 Provide help information
  load                 Load information from file
......

ibm@J48E> clear ?
Possible completions:
  arp                  Clear address resolution information
  auto-configuration   Clear auto-configuration action
  bfd                  Clear Bidirectional Forwarding Detection information
  bgp                  Clear Border Gateway Protocol information
  captive-portal       Clear 802.1X session
  chassis              Clear chassis information
```

The help topic command

You can use the **help** command in several ways. The **help topic** command shows usage guidelines for a statement as shown in Example 6-12.

Example 6-12 The help topic command

```
ibm@J48E> help topic interfaces ?
Possible completions:
  accept-data          Accept packets destined for virtual address
  accept-source-mac    Policers for specific source MAC addresses
  access-profile-chap  CHAP profile associated with physical interface
  accounting           Packet counting for transit traffic
  accounting-profile   Accounting profile
  acfc                 Compression of Address and Control fields in PPP header
  acknowledge-retries  Setting for link acknowledgment messages
  acknowledge-timer    Setting for link acknowledgment messages
  action-red-differential-delay  Setting for link services differential delay
  address              Interface address and destination prefix

......

ibm@J48E> help topic interfaces address
                       Configuring the Interface Address
```

```
   You assign an address to an interface by specifying the address when
   configuring the protocol family. For the inet family, configure the
   interface"s IP address. For the iso family, configure one or more
   addresses for the loopback interface. For the ccc, tcc, mpls, tnp, and
   vpls families, you never configure an address.
......
```

The help reference command

The **`help reference`** command shows summary information for a statement as shown in Example 6-13.

Example 6-13 The help reference command

```
ibm@J48E> help reference interfaces address
                                   address

    Syntax

   address address {
           arp ip-address (mac | multicast-mac) mac-address <publish>;
           broadcast address;
           destination address;
.....
Hierarchy Level

   [edit interfaces interface-name unit logical-unit-number family family],

   [edit logical-systems logical-system-name interfaces interface-name unit
   logical-unit-number family family]

    Release Information

   Statement introduced before JUNOS Release 7.4.

    Description

   Configure the interface address.
```

6.2.4 CLI configuration mode

You configure Junos software by entering configuration mode. Within configuration mode, you can configure all features of Junos software, such as interfaces, routing protocols, user access, routing policy, and chassis properties.

Active and candidate configuration

Junos software has two configuration files: *candidate configuration* and *active configuration*. The active configuration is the current running configuration in Junos software. The candidate configuration is the temporary text file that is being modified while in configuration mode.

The candidate configuration becomes the active configuration when the **`commit`** command is issued and no syntax errors are detected. The old active configuration is saved as a file called `rollback 1`.

In case a mistake occurs in the changes, you can easily recover the old active configuration by issuing the **`rollback 1`** command. The old active configuration replaces the candidate configuration. You must issue a **`commit`** command to activate the rollback file.

Junos software saves the last active configuration and the previous 49 configurations. Every time a **`commit`** command is issued, the saved file moves down to the list of 49. The first commit creates rollback1. The second commit becomes rollback 1, and the old rollback then becomes rollback 2. Figure 6-9 illustrates the rollback process.

Figure 6-9 Configuration rollback process

Entering configuration mode

To configure the router, enter configuration mode by typing the word `configure` or `edit` in operation mode. The router prompt changes to the # character as shown in Example 6-14.

Example 6-14 Entering configuration mode

```
ibm@J48E> configure
Entering configuration mode

{master:0}[edit]
ibm@J48E#

.....

ibm@J48E> edit
Entering configuration mode

{master:0}[edit]
ibm@J48E#
```

With *configure exclusive,* only a single user can configure the router. In exclusive mode, no other users can change the configuration besides the single user that entered exclusively.

With *configure private*, multiple users can configure different pieces of the configuration. Using private mode, each user gets a copy of the current configuration, and only changes that they make will be applied. If two users attempt to make the same change, such as changing an IP address to the same interface, the change is rejected, and both users exit the configuration mode to resolve their conflict.

Junos hierarchical configuration

The CLI is composed of many directories and subdirectories that build a hierarchical configuration. When you issue the **`set system services telnet`** command, the system directory is accessed, followed by the subdirectory services. You end the process with the **`telnet`** command to enable the Telnet service as shown in Figure 6-10.

Figure 6-10 Junos configuration hierarchy

Moving between levels

You can move between levels by using the **`edit`** command, which functions similarly to the change directory (**`cd`**) command. Figure 6-11 illustrates the move from the edit directory to the protocol subdirectory.

Figure 6-11 Changing directory with the edit command

Example 6-15 shows how to use the **edit** command to move the lower level of ospf.

Example 6-15 Moving to another directory

```
{master:0}[edit]
ibm@J48E# edit protocols ospf area 0 interface ge-0/0/0

{master:0}[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
ibm@J48E#
```

When in a certain directory, you can use multiple ways to navigate the directory tree with such commands as **up** and **top**, as shown in the following examples:

- You use the **up** command to move up one level in the hierarchy, as illustrated in Figure 6-12.

Figure 6-12 Moving up one level with the up command

Example 6-16 shows moving up one level by using the **up** command.

Example 6-16 Moving up one level

```
{master:0}[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
ibm@J48E# up

{master:0}[edit protocols ospf area 0.0.0.0]
ibm@J48E#
```

- You can use the **up n** command to move up *n* levels in the hierarchy as illustrated in Figure 6-13.

Figure 6-13 Moving up two levels

Example 6-17 shows how to move up two levels by using the **up n** command.

Example 6-17 Moving up two levels

```
{master:0}[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
ibm@J48E# up 2

{master:0}[edit protocols ospf]
ibm@J48E#
```

- The **top** command moves to the top of the hierarchy as illustrated in Figure 6-14.

Figure 6-14 Moving to the top of the hierarchy

Example 6-18 shows moving to the top of the hierarchy using the **top** command.

Example 6-18 Moving to the top of the hierarchy

```
{master:0}[edit protocols ospf area 0.0.0.0 interface ge-0/0/0.0]
ibm@J48E# top

{master:0}[edit]
ibm@J48E#
```

The delete command

The opposite of the **set** command to remove configuration from the router is the **delete** command. Usually this command is used to remove a single line. You can also use it to remove an entire hierarchy.

For example, to remove the Telnet service from the router, change the previous **set** command to a **delete** command shown in Example 6-19.

Example 6-19 Deleting a configuration

```
ibm@J48E# delete system services telnet
```

Viewing a candidate configuration

To view the candidate configuration, use the **show** command. This command shows the configuration at the current hierarchy level or at the specified level below the current location.

For example, you can view only the portions that you want to view from the root hierarchy as shown in Example 6-20.

Example 6-20 Showing a candidate configuration

```
ibm@J48E# show system services
ftp;
ssh;
telnet;

[edit]
ibm@J48E#
```

The commit command

The router and switch do not automatically apply the configuration. You must use the **commit** command to activate your candidate configuration.

Several options are possible for the **commit** command:

commit	Commit the current configuration
commit check	Check correctness of syntax but do not apply changes
commit confirmed	Temporarily active the configuration
commit at	Schedule to activate the changes at the specified time
commit comment	Add comments to the commit
commit and-quite	Activate the changes and exit configuration mode in a single step

The rollback command

The Junos software saves the last 50 committed versions of the configuration. To overwrite or restore the candidate configuration with one of these previously committed configurations, use the **rollback** command:

- Use the **rollback** or **rollback 0** command to reset the candidate configuration to the currently active configuration.
- Use the **rollback 1** command to load the previously active configuration.
- Use the **rollback n** command (**n** can be a number in the range 0 through 49) to load the referenced rollback version of the configuration.

The **rollback** command modifies only the candidate configuration. You must issue a **commit** command to activate the changes.

Saving a configuration

You can save the candidate configuration to an ASCII file. The file is saved to user's home directory. You can provide the full path name if you prefer to save the file in another directory.

The **save** command only saves the configuration in its current hierarchy (and below) along with the statement hierarchy that contains it. To save the entire candidate configuration, you must issue the **save** command at the top level of the configuration hierarchy.

Save a file by using a meaningful file name as shown in Example 6-21.

Example 6-21 Saving a configuration

```
ibm@J48E# save config-backup-2010-04-29
Wrote 261 lines of configuration to 'config-backup-2010-04-29'

[edit]
ibm@J48E#
```

Loading a configuration

You can retrieve the previously saved configuration by using the **load** command. The **load** command has several options for specific details of the operation.

load override filename	Completely overrides an existing configuration.
load merge filename	Combines the current configuration with the configuration being loaded.
load replace filename	Replaces existing statements in the current configuration.
load factory-default	Loads the factory-default configuration.

After load operation is completed, you must run a **commit** command to activate the changes of the configuration.

Comparing configurations

Junos software has a useful pipe command, **compare** command, that you can use to compare two files. With this command, any two files, including rollback files, active files, and candidate files, can be compared so that you can see the differences.

For example, the candidate and active configurations can be compared as shown in Example 6-22.

Example 6-22 Comparing the active and candidate configurations

```
ibm@J48E# show | compare
[edit system]
+  tacplus-server {
+      10.1.1.1;
+  }
[edit system services]
-    ftp;
+    ssh;

[edit]
```

You can also compare the active configuration with rollback configurations or saved files as shown in Example 6-23.

Example 6-23 Comparing an active configuration with rollback and saved files

```
[edit]
ibm@J48E> show configuration| compare rollback number

.....

ibm@J48E> show configuration| compare filename
```

The run command

In configuration mode, you can run operational-mode commands by using the **run** command. This command saves you time from existing to operational mode and entering configuration mode during configuration and verification.

You can use the **run ping** command from the interface directory in configuration mode as shown in Example 6-24.

Example 6-24 The run ping command

```
{master:0}[edit interfaces ge-0/0/0 unit 0]
ibm@J48E# run ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1): 56 data bytes
64 bytes from 192.168.1.1: icmp_seq=0 ttl=64 time=0.151 ms
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.111 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.297 ms
^C
--- 192.168.1.1 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.111/0.186/0.297/0.080 ms

{master:0}[edit interfaces ge-0/0/0 unit 0]
ibm@J48E#
```

Helpful configuration-mode commands

You can perform the following functions with the help of several commands that save time and increase accuracy and consistency during configuration:

- Rename a configuration statement.
- Replace a pattern of configuration statements.
- Copy a configuration statement to another statement.
- Deactivate a configuration statement.
- Insert a configuration statement into another location.

6.3 More information

For more information, see the following reference material:

- CLI reference material

  http://www.juniper.net/techpubs/en_US/junos10.1/information-products/topic-collections/swconfig-cli/swconfig-cli-TOC.html

- *J-Web Interface User Guide*

  http://jnpr.net/techpubs/en_US/junos10.1/information-products/topic-collections/jweb-user-guide/frameset.html

7

Virtual Chassis

With the IBM Ethernet Switch J48E, you can connect multiple IBM Ethernet Switch J48E to form one unit and manage the unit as a single chassis. The chassis is called a *Virtual Chassis*.

This chapter describes the following topics:

- Overview of a Virtual Chassis
- Deployment options for a Virtual Chassis
- Installing a Virtual Chassis configuration
- Management of a Virtual Chassis
- Operational monitoring of a Virtual Chassis

© Copyright IBM Corp. 2011. All rights reserved.

7.1 Overview of a Virtual Chassis

A Virtual Chassis is a collection of interconnected IBM Ethernet Switch J48Es that operate similar to a single chassis system and are managed as a single switch. A Virtual Chassis can consist of 1–10 IBM Ethernet Switch J48Es, providing high port density up to a total of 480 10/100/1000 fixed Ethernet ports.

In addition to the fixed ports, a Virtual Chassis offers up to forty 1 Gigabit Ethernet (SFP) uplink ports or twenty 10 Gigabit Ethernet (XFP) uplink ports.

7.1.1 Benefits of a Virtual Chassis

A Virtual Chassis has the following key benefits among others:

- Simplified management: Single management interface, single Junos software version, single copy of configuration, and chassis-like slot, module, and port numbering scheme
- Simplified network design: Single network entity, single control plane, and link aggregation across Virtual Chassis members
- Superior resiliency: Redundant master and backup Route Engines, redundant switch backplane, redundant power, and redundant cooling
- Superior performance at lower entry price point: Simplified distributed forwarding switch architecture, low-power consumption, and compact form factor
- Flexibility: The ability to add more Virtual Chassis members as port density grows, add more 10 Gigabit Ethernet uplinks, and mix and match switch types

7.1.2 Components of a Virtual Chassis

With a Virtual Chassis, you can interconnect 2–10 IBM Ethernet Switch J48Es and run them as a single network entity. You need at least two interconnected switches to take advantage of the Virtual Chassis feature. Any IBM Ethernet Switch J48E in a Virtual Chassis has some Virtual Chassis components.

Virtual Chassis ports

Two dedicated Virtual Chassis ports (VCPs) are on the rear panel of the IBM Ethernet Switch J48Es. They are used exclusively to interconnect IBM Ethernet Switch J48Es in a Virtual Chassis configuration. The interfaces for these dedicated ports are not configurable and operational by default when the ports are properly cabled. VCPs do not have cabling dependencies, which means that VCP 0 of a switch can be connected to VCP 0 or VCP 1 of another switch.

Figure 7-1 shows the Virtual Chassis ports on the IBM Ethernet Switch J48E rear panel.

Figure 7-1 Virtual Chassis ports on an IBM Ethernet Switch J48e rear panel

A proprietary Virtual Chassis port cable is used to interconnect the member switches and is included with each J48E switch package. The length of cable is 0.5 meter. Longer cables are available (1.5 meters, 3 meters, and 5 meters) that can be purchased separately.

Figure 7-2 shows the front and rear views of five J48E switches that are connected using a Virtual Chassis port cable.

Figure 7-2 Virtual Chassis configuration

IBM Ethernet Switch J48Es can be interconnected in an extended Virtual Chassis configuration across distances of up to 50 km (31.0685 mi) with an optional installation of either of the following uplink modules:

- IBM 45W4876 four-port 10BASE-X SFP uplink module
- IBM 45W4453 two-port 10 Gigabit Ethernet XFP uplink module

Ports on the uplink module are then configured to function as Virtual Chassis ports so that the interconnected switches are recognized as members of the same Virtual Chassis configuration. Multiple uplinks can be used to interconnect extended Virtual Chassis configurations for increased bandwidth and path redundancy.

With Junos Software Version 9.6, up to four Gigabit-Ethernet uplink-module ports or two 10 Gigabit-Ethernet uplink-module ports configured as Virtual Chassis ports can be combined to form a single point-to-point connection called a *link aggregation group (LAG)* or bundle. A LAG provides more bandwidth than a single link. Creating a LAG over Virtual Chassis ports creates a bigger “pipe” for forwarding traffic between Virtual Chassis member switches connected by those ports and greater operational redundancy.

Figure 7-3 shows the uplink modules in the front panel of an IBM Ethernet Switch J48E.

Figure 7-3 Virtual Chassis ports on an IBM Ethernet Switch J48E front panel

Each switch (non-Virtual Chassis port) port on a Virtual Chassis number is numbered *x*/*y*/*z*, where:

- *x* is the member ID of the member.
- *y* is the port interface control (PIC) ID. Uplink module ports are always on PIC 1, and fixed ports are always on PIC 0.
- *z* is the port number on the uplink or fixed PIC.

For example, Figure 7-4 shows three member switches in a Virtual Chassis configuration with the respective port number indicated.

Figure 7-4 Port numbering in a Virtual Chassis

Virtual Chassis member role

Each member switch in a Virtual Chassis configuration is assigned a specific role. A role determines the functions that the member performs in the configuration.

One member is assigned the master role and is responsible for managing other members in the Virtual Chassis configuration. Another member is assigned a backup role and takes over the master role if the master switch fails. All other members are assigned the linecard role. The system runs a mastership election algorithm to determine member roles in a Virtual Chassis.

Figure 7-5 shows member roles in a Virtual Chassis configuration.

Figure 7-5 Member roles in a Virtual Chassis configuration

Master role

The member switch that operates in the master role in a Virtual Chassis configuration has full control over the Virtual Chassis configuration. It performs the following functions:

- Serves as the active Routing Engine for the Virtual Chassis configuration.
- Manages the member switches in the Virtual Chassis configuration.
- Runs Junos for the Virtual Chassis configuration.
- Runs the chassis management processes and network control protocols.
- Calculates and maintains the forwarding table and distributes it to the local processor and then to Packet Forwarding Engines in all member switches.
- Receives and transmits routing information.
- Represents all member switches. (The host name and other properties that are assigned to the master switch during setup apply to all members of the Virtual Chassis configuration.)
- Holds the active and master copy of the entire Virtual Chassis configuration. Copies of the active configuration can also be synchronized to all member switches by using the commit sync command-line interface (CLI) command.

When an IBM Ethernet Switch J48E is powered on as a stand-alone switch, it is considered the master member. In a multimember Virtual Chassis configuration, one member functions as the master, and a second member functions as the backup.

Backup role

The member switch that functions in the backup role in a Virtual Chassis configuration performs the following functions:

- Serves as the backup Routing Engine for the Virtual Chassis configuration.
- Maintains synchronization with the master switch so that it can take over the master role if the master fails.

- Runs Junos for IBM Ethernet Switch J48Es in a backup role.
- Synchronizes with the master switch protocol states, forwarding table, and other configurations. By doing so, it is prepared to preserve forwarding information and maintain network connectivity with no or minimal disruption if the master switch becomes unavailable.

You must have at least two member switches in a Virtual Chassis configuration to have a backup member.

Linecard role

Each member that functions in the linecard role in a Virtual Chassis configuration performs the following functions:

- Runs Junos for IBM Ethernet Switch J48Es in a linecard role.
- Detects switch error conditions, such as an unplugged cable, on any interfaces that have been configured on it through the master switch and relays this information to the master switch.
- Receives updates about forwarding information from the master switch and programs these updates into the local Packet Forwarding Engine.

A member in a linecard role does not run full network control protocols. If a master or backup switch fails, one of the line card switches takes over the backup role. A Virtual Chassis configuration must have at least three members to include a line card member.

Member ID numbering

When an IBM Ethernet Switch J48E powers on, it receives a member ID, which is displayed on its front-panel LCD. If the switch powers on as a stand-alone switch, its member ID is 0. When the switch interconnects with other IBM Ethernet Switch J48Es in a Virtual Chassis configuration, the master switch assigns a member ID (0-9) to the switch. The member ID is based on various factors, including the order in which the switch was added to the Virtual Chassis configuration. As each switch is added and powered on, it receives the next lowest available (unused) member ID.

The member ID distinguishes the member switches from each other. You use the member ID in the following ways:

- Assign a mastership priority value to a member switch.
- Configure interfaces for a member switch (a function that is similar to a slot number on an IBM Ethernet Switch J08E or IBM Ethernet Switch J16E modular platform).
- Apply various operational commands to a member switch.
- View the status or characteristics of a member switch and its ports.

Member location: Members can be physically located in any order in a Virtual Chassis configuration. They do not need to be placed in order of member ID.

Figure 7-6 illustrates member ID assignments in a Virtual Chassis configuration with five members.

Figure 7-6 Member ID assignments in a Virtual Chassis

Member ID assignments follow these guidelines:

- When a Virtual Chassis member reboots, it retains its member ID.
- The member ID of a switch that is removed from a Virtual Chassis configuration is not released automatically to the available member ID pool.
- If the Virtual Chassis configuration previously included a member switch and that member was physically disconnected or removed from the Virtual Chassis, its member ID is not automatically available for re-assignment by the master switch. For example, consider a Virtual Chassis configuration that is composed of member 0, member 2, and member 3 because member 1 was removed. When you add another member switch and power it on, the master switch assigns it a member ID of 4.
- A replacement switch is treated as a new addition to the Virtual Chassis configuration and receives the next lowest available member ID.
- You can use the operational mode CLI command shown in Example 7-1 to manually configure the member ID of a member switch.

Example 7-1 The renumber member ID command

```
ibm@J48E> request virtual-chassis renumber member-id <current-member-id>
new-member-id <new-member-id>
```

- You can use the CLI command in Example 7-2 to return a member ID that was previously used, but is no longer assigned to any active member to the member ID pool.

Example 7-2 The reuse member ID command

```
ibm@J48E> request virtual-chassis recycle member-id <current-member-id>
```

Mastership priority setting

In a configuration that is not *preprovisioned*, the master, backup, or linecard role can be designated by configuring its mastership priority from 1 to 255. (*Preprovisioned* refers to an installation that uses a dynamic method.) In this case, the role is a member switch that performs within the Virtual Chassis configuration.

When an IBM Ethernet Switch J48E powers on, it receives the default mastership priority value of 128. A stand-alone IBM Ethernet Switch J48E can be connected to an existing Virtual Chassis configuration, which implicitly includes its own master. In this case, you must explicitly configure the mastership priority of the members that you want to function as the master and backup switches.

Use the following guidelines for assigning mastership priority:

- Specify the same mastership priority value for the master and backup switches in a Virtual Chassis configuration. Specifying the same value helps to ensure a smooth transition from master to backup if the master switch becomes unavailable. This configuration also prevents the original master switch from retaking control from the backup switch when the original master switch comes back online. This situation is sometimes referred to as *flapping* or *preemption* and can reduce the efficiency of system operation.
- Configure the highest possible mastership priority value (255) for the master and backup switches. With this configuration, the members continue to function as the master and backup switches when new members are added to avoid a mastership election process in the existing Virtual Chassis configuration.

Mastership election process

When a Virtual Chassis configuration starts, the Junos operating system automatically runs the mastership election process. This process determines which member switches take the role of master, backup, and line cards. All Virtual Chassis member switches participate in the election process. If a master switch fails, the backup switch automatically and immediately takes on the master role, which minimizes interruption to the operation of Virtual Chassis configuration. The system then runs the mastership election process to elect one of the line card switches as the new backup switch. The system also runs this process if the backup switch fails.

The election algorithm uses the following sequence to assign member roles and elect master and backup switches. The master role is assigned to the switch with the highest ranking. The backup role is assigned to the switch with the second highest ranking. Other switches become line cards.

1. Select the member with the highest user-configured mastership priority. (One is the lowest, and 255 is the highest possible value.)
2. Select the member that was the master switch the last time the Virtual Chassis configuration started.
3. In a Virtual Chassis configuration merge scenario, select the master member that has the highest number of current members in the existing Virtual Chassis configuration. A merge scenario occurs when two active Virtual Chassis configurations, each with its own master, are combined.

4. Select the member that has been included in the Virtual Chassis configuration for the longest time. For this factor to be considered, a lapse of at least one minute is required between the times that each interconnected member switch is started.
5. Select the member with the lowest Media Access Control (MAC) address.

To ensure that a specific member is elected as the master switch during initial Virtual Chassis installation, follow these steps:

1. Power on only the switch that you want to configure as the master switch of the Virtual Chassis configuration.
2. Configure the mastership priority of that member to have the highest possible value (255).
3. Without connecting to the master switch, power on all other member switches individually. Reset their configurations to the factory default values by using either the front LCD menu or a CLI command.
4. Connect all other members to the master switch while they are powered off.
5. Power on the other member switches.
6. Continue to configure other member switches through the master switch, as desired. These member switches default to a mastership priority of 128 if no previous configuration exists.

Software compatibility check

A Virtual Chassis configuration can include IBM Ethernet Switch J48Es with different hardware versions and model numbers. However, each member switch must run the same software version as the master switch. When a member switch is added to a Virtual Chassis configuration, the master switch checks the compatibility of the Junos operating system running on the new switch. A switch running a different software version than the master switch obtains a member ID from the master switch. However, the switch does not become a functional member of the Virtual Chassis configuration and does not forward data packets.

You must download the required software package to the master switch in the Virtual Chassis configuration. Then you can upgrade the new member switch to the stored software version by using the CLI command shown in Example 7-3.

Example 7-3 Software upgrade from the master switch

```
ibm@J48E> request system software add <package location> member <member-id> reboot
```

This command downloads the image from the master switch through the Virtual Chassis ports to the specified member and then reboots the member. The member does not need to be directly connected to the master switch.

With Junos operating system Release 10.0, the automatic software upgrade feature (Example 7-4) is available. This feature automatically upgrades the software version on new member switches when joining a Virtual Chassis configuration to the same version of the Junos operating system running on the master switch. This feature is disabled by default. You must explicitly enable it and specify a path, either to a local directory or a remote server location, where new Virtual Chassis switch members can obtain the appropriate software image.

Example 7-4 Automatic software upgrade by master switch

```
{master:0}[edit]
root# edit virtual-chassis

{master:0}[edit virtual-chassis]
root# set auto-sw-update package-name </image-path>
```

7.1.3 Cabling options for a Virtual Chassis

Physical placement of IBM Ethernet Switch J48Es in a Virtual Chassis configuration is flexible. The switches can be located in a same rack or across multiple racks in the same or different telecommunications closet in floors or buildings.

When interconnecting the switches through dedicated Virtual Chassis ports, the maximum Virtual Chassis port cable length is 5 meters. If you are extending the Virtual Chassis configuration beyond this length, uplink ports offer a greater distance to interconnect two switches in different locations.

Connect switches in a Virtual Chassis configuration in a ring topology for resiliency and speed. A ring configuration provides up to 128 Gbps of bandwidth between member switches.

You can interconnect switches in a Virtual Chassis configuration by using one of three cabling methods, which are described in the following sections.

Daisy-chained ring configuration

In the daisy-chained ring configuration, each device in the Virtual Chassis configuration is connected to the device immediately next to it. Also, members at the end of the Virtual Chassis configuration are connected to each other to complete the ring topology. Connections between devices can use either Virtual Chassis port on the back of a device (for example, VCP 0 to VCP 0 or VCP 0 to VCP 1).

The daisy-chained ring configuration provides a simple and intuitive method for interconnecting devices. The maximum height or breadth of the Virtual Chassis is 5 meters as shown in Figure 7-7.

Figure 7-7 Daisy-chained ring cabling

Braided ring configuration

In the braided ring cabling configuration, alternating devices in the Virtual Chassis configuration are connected to each other. In addition, the two device pairs at each end of the Virtual Chassis configuration are directly connected to each other to complete the ring topology. Connections between devices can use either Virtual Chassis port on the back of a device.

The braided ring configuration extends the Virtual Chassis height or breadth to 22.5 meters (73.8188 ft) as shown in Figure 7-8.

Figure 7-8 Braided ring cabling

Extended Virtual Chassis configuration

With the extended Virtual Chassis configuration, individual Virtual Chassis members or dedicated Virtual Chassis configurations can be interconnected across distances of up to 50 km (31.0685 mi) with redundant fiber links. In this configuration, you can use uplink modules, such as the following examples, to interconnect the members of the Virtual Chassis configuration:

- IBM 45W4876 four-port 10BASE-X SFP uplink module
- IBM 45W4453 two-port 10 Gigabit Ethernet XFP uplink module

Multiple uplinks can be grouped together for additional bandwidth and link redundancy.

Each member switch in this configuration must be configured to enable uplink ports as Virtual Chassis ports before interconnecting with each other as shown in Example 7-5.

Example 7-5 Configuring an uplink port

```
ibm@J48E> request virtual-chassis vc-port set pic-slot <pic-slot> port <port>
member <member-id>
```

Figure 7-9 shows the extended Virtual Chassis configuration.

Figure 7-9 Extended Virtual Chassis cabling

7.2 Deployment options for a Virtual Chassis

The following sections provide overviews of various deployments of the Virtual Chassis configuration in data centers.

7.2.1 Top-of-rack deployment in a data center

The top-of-rack deployment in a data center is suitable for data center environments in which members of the Virtual Chassis configuration are colocated with servers in the same rack. A single Virtual Chassis configuration that consists of multiple switches in the same rack simplifies management by reducing the number of logically managed devices. It offers flexible options for the number and deployment of uplinks. This deployment also provides servers with the ability to configure NIC teaming (LAG) to multiple members of the same Virtual Chassis configuration. The benefit is increased total server network bandwidth and server link redundancy.

Figure 7-10 illustrates top-of-rack deployment in a data center in a single rack.

Figure 7-10 Top-of-rack deployment in a data center in a single rack

The Virtual Chassis configuration can be extended across multiple racks within the same row in a data center. Figure 7-11 shows the top-of-rack deployment in a data center extended across several racks. It illustrates a daisy-chained, ring cabling configuration, which allows spacing of up to 5 meters (16.4041 ft) between adjacent racks and requires one or more uplink extensions to complete the ring. Extending the Virtual Chassis across racks further eases management, lowers inter-rack server-to-server latency, and increases uplink flexibility, ultimately leading to lower total cost of ownership (TCO).

Figure 7-11 Top-of-rack deployment in a data center with daisy-chained ring cabling configuration (single switch per rack shown for simplification)

Figure 7-12 shows an example of extending the top-of-rack deployment in a data center across several racks. It illustrates a braided-ring cabling configuration, which allows spacing of up to 5 meters (16.4041 ft) between alternate racks.

Figure 7-12 Top-of-rack deployment in a data center with braided-ring cabling configuration (single switch per rack shown for simplification)

The Virtual Chassis configuration can be extended across multiple racks and across rows in a data center, as shown in Figure 7-13.

Figure 7-13 Top-of-rack deployment in a data center with multiple racks and rows (single switch per rack shown for simplification)

7.2.2 End-of-row deployment in a data center

The end-of-row deployment in a data center allows the installation of IBM Ethernet Switch J48Es as end-of-row devices in a data center. Figure 7-14 illustrates this deployment. Virtual Chassis members are interconnected by using the daisy-chained or braided-ring cabling configuration, and uplinks provide connectivity to core switching devices.

Figure 7-14 End-of-row deployment in a data center (single switch per rack shown for simplification)

7.3 Installing a Virtual Chassis configuration

The following sections describe options for installing a Virtual Chassis configuration. Two methods are possible for installing and configuring Virtual Chassis technology:

- Dynamic method (nonprovisioning)

  The dynamic method offers a simple plug-and-play option for building a Virtual Chassis configuration. It does not require any configuration. However, it does not prevent certain user errors, such as adding the wrong switch into a Virtual Chassis configuration.

- Preprovisioning method

  The preprovisioning method requires careful planning and user configuration before installing the Virtual Chassis configuration. Because all member switches and their roles in a given Virtual Chassis must be statically defined, this method minimizes user error and provides consistent and deterministic results if a member switch fails.

7.3.1 Building a new Virtual Chassis with the dynamic method

Use the dynamic installation method to build a Virtual Chassis configuration without prior user configuration. Although it is not required, configure the master and backup switches to have the highest value of mastership priority (255) to avoid the mastership election process whenever a new switch is added with a higher mastership priority.

For best practice, always reset all member switches to the factory default configuration before adding these switches to the Virtual Chassis configuration. This procedure prevents unexpected behavior during the addition of the new member. Factory defaults can be loaded as shown in Example 7-6.

Example 7-6 Loading factory defaults on a switch

```
root@J48E# load factory-default
warning: activating factory configuration

{master:0}[edit]
root@J48E# set system root-authentication plain-text-password
New password:
Retype new password:

{master:0}[edit]
root@J48E# commit
configuration check succeeds
commit complete
```

LCD menus: In addition to using CLI commands, you can use LCD menus on a switch to load the default configuration from the factory.

Figure 7-15 shows a Virtual Chassis of three IBM Ethernet Switch J48Es. The setup of this Virtual Chassis configuration uses the dynamic method.

Figure 7-15 A Virtual Chassis of three IBM Ethernet Switch J48Es running in dynamic mode

To set up this Virtual Chassis configuration using the dynamic method, follow these steps:

1. Set up an IBM Ethernet Switch J48E as a master switch (RE0):
   a. Install the required power supplies.
   b. Place the switch in the desired location.
   c. If the new switch was previously configured, reset the configuration to the factory default values.
   d. Power up the switch.

      When logging in the switch, you see that the switch becomes the master (RE0) and has a member ID of 0 as shown in Example 7-7.

      Example 7-7 A master switch with a member ID of 0

```
Amnesiac (ttyu0)

login: root

Logging to master


--- JUNOS 10.1R1.8 built 2010-02-12 17:24:20 UTC

root@:RE:0% cli
{master:0}
root>

{master:0}
```

   e. Assign the highest mastership priority to the switch as shown in Example 7-8.

      Example 7-8 Assigning the mastership priority to a master switch

```
{master:0}[edit virtual-chassis]
root# set member 0 mastership-priority ?
Possible completions:
  <mastership-priority>  Member's mastership priority (1..255)
{master:0}[edit virtual-chassis]
root# set member 0 mastership-priority 255
```

2. Set up an IBM Ethernet Switch J48E as a backup switch (RE1):
   a. Install the required power supplies.
   b. Power up the switch and load the factory default configuration if the switch was previously configured.
   c. Power off the switch.
   d. Place the switch in the desired location.
   e. Connect the switch to existing members of the Virtual Chassis configuration with a Virtual Chassis port cable.
   f. Power up the switch.

      The mastership election process assigns the switch the next lowest available member ID based on the order that the switch is added to the Virtual Chassis configuration.

g. Use the **show virtual-chassis status** command to view the information. In this setup, the switch is assigned member ID of 1 with a default mastership priority as shown in Example 7-9.

Example 7-9 Viewing the status of the backup switch in a Virtual Chassis

```
root> show virtual-chassis status

Virtual Chassis ID: bc80.d169.c61f
                                          Mastership            Neighbor
List
Member ID  Status   Serial No    Model    priority    Role      ID
Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      255  Master*    1  vcp-0
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t      128  Backup     0  vcp-0

Member ID for next new member: 2 (FPC 2)

{master:0}
```

h. Assign the backup switch the same mastership priority as the master switch, as shown in Example 7-10, which prevents mastership preemption if the master fails and then recovers.

Example 7-10 Assigning mastership priority to the backup switch

```
{master:0}[edit]
root# edit virtual-chassis

{master:0}[edit virtual-chassis]
root# set member 1 mastership-priority 255
```

3. Set up an IBM Ethernet Switch J48E as a line card switch:
   a. Install the required power supplies.
   b. Power up the switch and load the factory default configuration if the switch was previously configured.
   c. Power off the switch.
   d. Place the switch in the desired location.
   e. Connect the switch to the existing members of the Virtual Chassis with a Virtual Chassis port cable.
   f. Power up the switch.

      The mastership election process assigns the switch the next lowest available member ID based on the order in which the switch is added to the Virtual Chassis configuration.
   g. Use **show virtual-chassis status** command to view the information as shown in Example 7-11. In this setup, it is assigned a member ID of 2.

Example 7-11 Viewing the line card switch in a Virtual Chassis

```
root> show virtual-chassis status

Virtual Chassis ID: bc80.d169.c61f
                                          Mastership            Neighbor
List
```

```
Member ID  Status   Serial No    Model    priority    Role      ID
Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t     255  Master*    1  vcp-0
                                                               2  vcp-1
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t     255  Backup     0  vcp-0
2 (FPC 2)  Prsnt    134002W      ex4200-48t     128  Linecard   0  vcp-1

Member ID for next new member: 3 (FPC 3)

{master:0}
```

7.3.2 Building a new Virtual Chassis with the preprovisioning method

With a preprovisioned configuration, you can deterministically control the member ID and role assigned to a member switch by associating the switch to its serial number. A preprovisioned configuration file links the serial number of each IBM Ethernet Switch J48E to a designated member ID and role. The serial number must be specified in the configuration file for the member to be recognized as part of the Virtual Chassis configuration.

In this configuration, you must select two members that you want to make eligible for election as the master and backup switches. When you list these two members in the preprovisioned configuration, you designate the member role as a Routing Engine. One member then functions as the master switch of the Virtual Chassis configuration, and the other functions as the backup switch.

Additional members, which are not eligible for election as the master or backup switch, can be specified as line cards in the preprovisioned configuration.

The preprovisioned configuration provides the option of not explicitly assigning a role to a member switch. This member switch is eligible for election as the backup switch if the master or the backup switch fails. It can also become the master switch if both the master and backup switches fail.

In a preprovisioned configuration, the following configurations are allowed:

- One member can be explicitly configured as the master switch, and one member can be explicitly configured as the backup switch.
- You can explicitly configure a member with the role of line card, which makes it ineligible for functioning as a master or backup switch.
- A member that is not explicitly assigned a role is eligible to become either of the following types of switches:
  - The backup switch (if the master switch or backup switch fails)
  - The master switch (if both the master switch and backup switches fail).

The mastership priority value is assigned by the switch software based on the specified role:

- The master and backup switches are assigned a mastership priority of 129.
- A line card switch is assigned a mastership priority of 0, making it ineligible to participate in the master election.
- A switch that is not explicitly assigned a role is configured with a mastership priority of 128, making it eligible to participate in the master election.

Figure 7-16 shows a Virtual Chassis of three IBM Ethernet Switch J48Es. The setup of this Virtual Chassis uses the preprovisioning method.

Figure 7-16 A Virtual Chassis of three IBM Ethernet Switch J48Es running in preprovisioned mode

Before you build the Virtual Chassis in preprovisioned mode, complete these prerequisite tasks:

- Make a list of the serial numbers of all the switches to be connected in a Virtual Chassis configuration.
- Determine the desired role (Routing Engine or line card) of each switch. If you configure the member with a Routing Engine role, it is eligible to function as a master or backup. If you configure the member with a linecard role, it is not eligible to become a master or backup.

To build a new Virtual Chassis with the preprovisioning method, follow these steps:

1. Set up the IBM Ethernet Switch J48Es as a master switch with preprovisioned mode:
   a. Install the required power supplies.
   b. Place the switch in the desired location.
   c. If the new switch was previously configured, reset the configuration to the factory default values.
   d. Power up the switch.
   e. When logging in the switch, specify the preprovisioned configuration mode as shown in Example 7-12.

   Example 7-12 Setting the pre-provisioned mode

   ```
   {master:0}[edit virtual-chassis]
   root# set preprovisioned
   ```

   f. Specify all the members that you want to included in the Virtual Chassis configuration. List the serial numbers of each switch with the desired member ID and role as shown in Example 7-13.

   Example 7-13 Setting the serial number and role of the member switches

   ```
   {master:0}[edit virtual-chassis]
   root# set member 0 serial-number BP0208174896 role routing-engine

   {master:0}[edit virtual-chassis]
   ```

```
root# set member 1 serial-number BP0208110806 role routing-engine

{master:0}[edit virtual-chassis]
root# set member 2 serial-number 134002W role line-card

{master:0}[edit virtual-chassis]
root# commit
configuration check succeeds
commit complete
```

g. After the **commit** command is issued, view the Virtual Chassis information by using the **show virtual-chassis status** command as shown in Example 7-14. You see that the switch is running in preprovisioning mode and it becomes a master switch. Also, its mastership priority is assigned 129 by the system.

Example 7-14 Viewing the master switch configuration in preprovisioned mode

```
root> show virtual-chassis status

Preprovisioned Virtual Chassis
Virtual Chassis ID: c3e9.bc7f.39ad
                                          Mastership            Neighbor
List
Member ID  Status   Serial No    Model    priority    Role      ID
Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      129  Master*

{master:0}
```

2. Set up an IBM Ethernet Switch J48E as a backup switch (RE1) with preprovisioned mode:
   a. Install the required power supplies.
   b. Power up the switch and load the factory default configuration if the switch was previously configured.
   c. Power off the switch.
   d. Place the switch in the desired location.
   e. Connect the switch to the existing members of the Virtual Chassis configuration with a Virtual Chassis port cable.
   f. Power up the switch.

   Similar to the preprovisioned mode where the serial number and role of member switches are configured during the master switch configuration, the switch is assigned its role based on the serial number by the master switch. Example 7-15 shows the backup switch configuration in a preprovisioned mode.

Example 7-15 Viewing the backup switch configuration in preprovisioned mode

```
root> show virtual-chassis status

Preprovisioned Virtual Chassis
Virtual Chassis ID: c3e9.bc7f.39ad
                                          Mastership            Neighbor
List
Member ID  Status   Serial No    Model    priority    Role      ID
Interface
```

```
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      129  Master*    1  vcp-0
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t      129  Backup     0  vcp-0
```

3. Set up an IBM Ethernet Switch J48E as a line card switch with preprovisioned mode:
   a. Install the required power supplies.
   b. Power up the switch and load the factory default configuration if the switch was previously configured.
   c. Power off the switch.
   d. Place the switch in the desired location.
   e. Connect the switch to existing members of the Virtual Chassis with a Virtual Chassis port cable.
   f. Power up the switch.

   With the serial number preset during master switch configuration, the switch is assigned as a line card switch, and its mastership priority is 0. This configuration makes it ineligible to be a master or backup switch as shown in Example 7-16.

Example 7-16 Viewing the line card switch configuration in preprovisioned mode

```
root> show virtual-chassis status

Preprovisioned Virtual Chassis
Virtual Chassis ID: c3e9.bc7f.39ad
                                          Mastership            Neighbor
List
Member ID  Status   Serial No    Model    priority    Role      ID
Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      129  Master*    1  vcp-0
                                                                 2  vcp-1
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t      129  Backup     0  vcp-0
2 (FPC 2)  Prsnt    134002W      ex4200-48t        0  Linecard   0  vcp-1

{master:0}
```

Mastership priority: You cannot modify the mastership priority when you are using a preprovisioned configuration. The mastership priority values are generated automatically and controlled by the role that is assigned to the member switch in the configuration file. The two Routing Engines are assigned the same mastership priority value. However, the member that was powered on first has higher prioritization according to the master election algorithm.

Example 7-17 shows that the mastership priority cannot be changed when running in preprovisioned mode.

Example 7-17 Attempted changing of the mastership priority in preprovisioned mode

```
{master:0}[edit virtual-chassis]
root# set member 0 mastership-priority 255

{master:0}[edit virtual-chassis]
root# commit
[edit virtual-chassis]
  'member 0'
    Can't have mastership priority configuration with preprovisioned set
```

```
error: configuration check-out failed

{master:0}[edit virtual-chassis]
```

7.3.3 Adding a new switch to an existing Virtual Chassis configuration in the same telecommunications closet

You can add a new switch to a Virtual Chassis configuration in the same telecommunications closet as the other member or members of the Virtual Chassis configuration. Figure 7-17 shows two IBM Ethernet Switch J48Es that form a Virtual Chassis. A new switch is added to the existing Virtual Chassis.

Figure 7-17 Adding a new switch to the existing Virtual Chassis

Before you add a new switch to an existing Virtual Chassis configuration in the same telecommunications closet, complete these prerequisite tasks:

- Install required power supplies in the new switch.
- Place the new switch in the desired location.
- Confirm that the new switch is powered off.
- If expanding a pre-provisioned configuration, make a note of the serial number on the back of the switch. Then edit the existing Virtual Chassis pre-provisioning configuration to include the serial number and the role of the new switch.

Add a new switch to the existing Virtual Chassis as explained in the following steps:

1. If the new member switch was previously configured, power it on, reset the configuration to the factory default values, and then power it off.
2. Interconnect the new switch to at least one member of the existing Virtual Chassis configuration by using the dedicated Virtual Chassis ports.
3. Power on the new switch.

4. Confirm that the new switch is now included within the Virtual Chassis configuration by issuing the **show virtual-chassis status** command as shown in Example 7-18. The new switch is added as member ID 2 (the next available member ID) and its role is line card.

Example 7-18 Viewing the status of the new switch in Virtual Chassis

```
root> show virtual-chassis status

Virtual Chassis ID: bc80.d169.c61f
                                          Mastership            Neighbor List
Member ID  Status   Serial No    Model    priority    Role      ID  Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      255  Master*    1  vcp-0
                                                                  2  vcp-1
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t      255  Backup     0  vcp-0
2 (FPC 2)  Prsnt    134002W      ex4200-48t      128  Linecard   0  vcp-1

Member ID for next new member: 3 (FPC 3)
```

If you are using a preprovisioned configuration, the member ID is assigned to the serial number of the member in the configuration file. For example, the new switch has a serial number of 134002W. You can configure the new switch to a member ID of 8 and a role of line card as shown in Example 7-19.

Example 7-19 Configuring the new switch to member ID 8 and a linecard role

```
{master:0}[edit]
root# set virtual-chassis member 8 serial-number 134002W

{master:0}[edit]
root# set virtual-chassis member 8 serial-number line-card

{master:0}[edit]
```

5. Confirm the new switch in the Virtual Chassis by entering the **show virtual-chassis status** command in preprovisioned mode as shown in Example 7-20. The new switch is added as member ID 8, and a role of line card is configured.

Example 7-20 Adding a new switch to the existing Virtual Chassis in preprovisioned mode

```
root# run show virtual-chassis status

Preprovisioned Virtual Chassis
Virtual Chassis ID: c3e9.bc7f.39ad
                                          Mastership            Neighbor List
Member ID  Status   Serial No    Model    priority    Role      ID  Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      129  Master*    1  vcp-0
                                                                  8  vcp-1
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t      129  Backup     0  vcp-0
8 (FPC 8)  Prsnt    134002W      ex4200-48t        0  Linecard   0  vcp-1

{master:0}[edit]
```

7.3.4 Adding a new switch from a remote telecommunications closet to an existing Virtual Chassis configuration

To add a new switch from a remote telecommunications closet to an existing Virtual Chassis configuration, you can use either of the following uplink modules:

- The IBM 45W4876 four-port 10BASE-X SFP uplink module
- The IBM 45W4453 two-port 10 Gigabit Ethernet XFP uplink module

Configure the uplink ports on both sides of the link as Virtual Chassis ports.

Figure 7-18 shows how two IBM Ethernet Switch J48Es form a Virtual Chassis in the same telecommunications closet. A new switch is added to the existing Virtual Chassis from a remote telecommunications closet.

Figure 7-18 Adding a new switch to the existing Virtual Chassis from remote telecommunications closet

Before you add a new switch from a remote telecommunications closet to an existing Virtual Chassis configuration, complete these prerequisite tasks:

- Install the required power supplies and uplink modules in the new switch.
- If the new switch was previously configured, reset the configuration to the factory default values.
- Power on the new switch as a stand-alone device to configure its uplink port as a Virtual Chassis port. Example 7-21 shows the configuration on port 0 of PIC 1. You must also configure an uplink port on one of the existing Virtual Chassis switches that is to be connected by the new switch.

Example 7-21 Configuring the uplink port as a Virtual Chassis port

```
{master:0}
root> request virtual-chassis vc-port set pic-slot 1 port 0

{master:0}
```

- After the uplink port is configured as Virtual Chassis port, you can verify it by issuing the **show virtual-chassis vc-port** command. Example 7-22 shows all Virtual Chassis ports of the new switch. Interface 1/0 is a Virtual Chassis port configured on the uplink port.

Example 7-22 Viewing the Virtual Chassis port status

```
{master:0}
root> show virtual-chassis vc-port
fpc0:
--------------------------------------------------------------------------
Interface   Type              Trunk  Status       Speed        Neighbor
or                             ID                 (mbps)       ID  Interface
PIC / Port
vcp-0       Dedicated           1    Down         32000
vcp-1       Dedicated           2    Down         32000
1/0         Configured         -1    Down         10000

{master:0}
```

- Place the new switch in the desired location.
- If you are expanding a preprovisioned configuration, note the serial number on the back of the switch. Then edit the existing Virtual Chassis configuration to include the serial number of the new member switch. Specify the member ID and the role of the new switch when adding its serial number in the Virtual Chassis pre-provisioned configuration. The parameters that are specified in the master Virtual Chassis configuration file are applied after the new member switch interconnects with its uplink Virtual Chassis port.
- Prepare an existing member for interconnecting with the new switch through an uplink port by configuring an uplink port as a Virtual Chassis port on the existing member.

To add the new switch, follow these steps:

1. Interconnect the new member switch to the existing Virtual Chassis configuration by using the uplink ports that have been configured as Virtual Chassis ports on both sides of the uplink.
2. Power on the new switch.
3. Confirm that the new switch is included within the Virtual Chassis configuration by issuing the **show virtual-chassis status** command. Example 7-23 shows a new switch included in the existing Virtual Chassis using an uplink port.

Example 7-23 Viewing the new switch in the Virtual Chassis using an uplink port

```
{master:0}
root> show virtual-chassis status

Virtual Chassis ID: 4bd4.cedc.d8e0
                                          Mastership            Neighbor List
Member ID  Status   Serial No    Model    priority    Role      ID  Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      255  Master*    1  vcp-0
                                                                 2  vcp-255/1/0
1 (FPC 1)  Prsnt    134002W      ex4200-48t      255  Backup     0  vcp-1
2 (FPC 2)  Prsnt    BP0208110806 ex4200-48t      128  Linecard   0  vcp-255/1/0

Member ID for next new member: 3 (FPC 3)

{master:0}
```

7.3.5 Replacing a member switch

You can replace a member switch in a Virtual Chassis configuration with little loss of traffic. When you replace a switch, the configuration of the replaced switch is retained and can be applied to the new switch if desired.

The following sections provide information about replacing a member switch in various situations.

> **Replacement switch configuration:** A replacement switch does not inherit the configuration of the previous switch. The switch has a different MAC address, and it will be assigned a different member ID.

Removing, repairing, and re-installing the same switch

If a member switch needs replacement, you can remove it from the Virtual Chassis configuration with little loss of traffic. The master switch stores the configuration, including the member ID of the removed switch, so that it can be reapplied when the switch (with the same MAC address) is reconnected.

To remove, repair, and reinstall a switch, follow these steps:

1. Power off and disconnect the switch.
2. Repair the switch, as necessary.
3. Reconnect and power on the switch.

In some cases, you must perform this procedure when the same switch has the following requirements:

- Change the power supply if it is running on a single power supply or installation.
- Change the new uplink module.

Removing a member switch, replacing it with a different switch, and re-applying the old configuration

You can replace a member switch with a different switch that retains the configuration of the original switch. The master switch stores the configuration of the member that was removed. When a replacement member switch is connected, the master switch assigns a new member ID to it. The old configuration remains stored under the member ID of the previous member switch. Renumbering the replacement switch with the member ID of the old switch applies the configuration of the old switch to the replacement switch.

Figure 7-19 shows a switch (SW-2) that is to be replaced with a new switch that keeps the same configuration of SW-2.

Figure 7-19 Replacing a different switch with the same configuration of a switch

Before you replace a switch and reapply configuration, view the current Virtual Chassis configuration. Example 7-24 shows two switches in the Virtual Chassis.

Example 7-24 Viewing the current Virtual Chassis before replacing the switch

```
{master:0}
root> show virtual-chassis

Virtual Chassis ID: 03a4.a19e.e10a
                                          Mastership            Neighbor List
Member ID  Status   Serial No    Model    priority    Role      ID  Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      255  Master*    1  vcp-0
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t      255  Backup     0  vcp-0

Member ID for next new member: 2 (FPC 2)

{master:0}
```

To replace a switch and reapply configuration, perform the following steps:

1. Power off and disconnect the switch to be replaced.
2. If the replacement switch has been previously configured, reset its configuration to the factory default values.

3. Connect the replacement switch in place of the old switch and power it on.

   The member ID that assigned to the replacement switch is the next lowest member ID that was available before the replacement. Example 7-25 shows the replacement switch's member ID of 2. Member ID 1 is not present because it is removed from the Virtual Chassis.

   Example 7-25 Viewing the replacement switch in a Virtual Chassis

```
{master:0}[edit]
root# run show virtual-chassis

Virtual Chassis ID: 03a4.a19e.e10a
                                          Mastership            Neighbor List
Member ID  Status   Serial No    Model    priority    Role      ID  Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      255  Master*    2  vcp-0
1 (FPC 1)  NotPrsnt BP0208110806 ex4200-48t
2 (FPC 2)  Prsnt    134002W      ex4200-48t      128  Backup     0  vcp-0

Member ID for next new member: 3 (FPC 3)

{master:0}[edit]
```

4. Change the replacement switch's member ID from 2 to 1 as shown in Example 7-26.

   Example 7-26 Renumbering the member ID of the replacement switch

```
{master:0}
root> request virtual-chassis renumber member-id 2 new-member-id 1

To move configuration specific to member ID 2 to member ID 1, please
use the replace command. e.g. replace pattern ge-2/ with ge-1/

Do you want to continue ? [yes,no] (no) yes


{master:0}
```

5. View the replacement switch change. Its member ID is now renumbered to 1 as shown in Example 7-27.

   Example 7-27 New member ID of the replacement switch

```
root> show virtual-chassis status

Virtual Chassis ID: 03a4.a19e.e10a
                                          Mastership            Neighbor List
Member ID  Status   Serial No    Model    priority    Role      ID  Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      255  Master*    1  vcp-0
1 (FPC 1)  Prsnt    134002W      ex4200-48t      255  Backup     0  vcp-0

Member ID for next new member: 2 (FPC 2)

{master:0}
```

Removing a member switch and making its member ID available for re-assignment to a different switch

When removing a member switch from the Virtual Chassis configuration, the master switch keeps the member ID of that switch on reserve. For example, a Virtual Chassis configuration has three members as shown in Example 7-28.

Example 7-28 Member ID in a Virtual Chassis

```
{master:0}
root> show virtual-chassi stauts

Virtual Chassis ID: 6140.f8bf.28a0
                                          Mastership            Neighbor List
Member ID  Status   Serial No    Model    priority    Role      ID  Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      255  Master*    1  vcp-0
                                                                  2  vcp-1
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t      255  Backup     0  vcp-0
2 (FPC 2)  Prsnt    134002W      ex4200-48t      128  Linecard   0  vcp-1

Member ID for next new member: 3 (FPC 3)

{master:0}
```

When member ID 2 is removed from the Virtual Chassis, its member ID is retained in the Virtual Chassis. However, as shown in Example 7-29, its status is *Not Present*, and it is not assigned to other new switches.

Example 7-29 Member ID 2 with a Not Present status

```
root> show virtual-chassis status

Virtual Chassis ID: 6140.f8bf.28a0
                                          Mastership            Neighbor List
Member ID  Status   Serial No    Model    priority    Role      ID  Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      255  Master*    1  vcp-0
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t      255  Backup     0  vcp-0
2 (FPC 2)  NotPrsnt 134002W      ex4200-48t

Member ID for next new member: 3 (FPC 3)

{master:0}
```

To reuse member ID 2, you can enter the **request virtual-chassis recycle member-id 2** command as shown in Example 7-30. This command frees member ID 2 for re-assignment of the next switch.

Example 7-30 Recycling the member ID of a Virtual Chassis

```
{master:0}
root> request virtual-chassis recycle member-id 2

{master:0}

*******
root> show virtual-chassis
```

```
Virtual Chassis ID: 6140.f8bf.28a0
                                          Mastership            Neighbor List
Member ID  Status   Serial No    Model    priority    Role      ID  Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      255  Master*    1  vcp-0
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t      255  Backup     0  vcp-0

Member ID for next new member: 2 (FPC 2)

{master:0}
```

7.4 Management of a Virtual Chassis

A Virtual Chassis configuration is composed of multiple IBM Ethernet Switch J48Es. It has multiple console ports and multiple out-of-band management Ethernet ports on the rear panels of the switches. Yet, a Virtual Chassis configuration requires single management.

7.4.1 Console port access

You can connect a PC or mobile computer directly to a console port of any member switch to set up and configure the Virtual Chassis. When you connect to the console port of any member switch, the console session is redirected to the master switch, as shown in Figure 7-20.

Figure 7-20 Console session redirected to a master switch

If the master becomes unavailable, the console session is disconnected from the old master, and a new session is established with the newly elected master.

7.4.2 Management Ethernet port access

An out-of-band management Ethernet port is often referred to as a *management Ethernet port*. It uses a dedicated management channel for device maintenance and allows a system administrator to monitor and manage the switch by remote control.

The Virtual Chassis configuration can be accessed and managed remotely using Secure Shell (SSH) or Telnet to a global management interface of a Virtual Chassis called the *Virtual Management Ethernet (VME) interface*.

VME is a logical interface that represents any of the out-of-band management ports on the member switches. When you connect to the Virtual Chassis configuration using the VME IP address, the connection is redirected to the master member as shown in Figure 7-21.

Figure 7-21 Management Ethernet port redirected to VME

If the master management Ethernet link is unavailable, the session is redirected through the backup management Ethernet link. If no active management Ethernet link is on the backup, the VME interface chooses a management Ethernet link on one of the line card members. It selects the line card member with the lowest member ID as its first choice.

7.4.3 Management IP address access

A Virtual Chassis configuration is managed as a single logical network element. Therefore, it has only one management IP address, which is configured on the VME interface. This VME interface is a logical IP interface associated with the Virtual Chassis internal management VLAN that connects the me0 interfaces of all member switches in a Virtual Chassis configuration. Example 7-31 shows how to assign an IP address.

Example 7-31 Configuring the IP address of the VME interface

```
{master:0}[edit]
ibm@J48E-VC# set interfaces vme unit 0 family inet address 10.1.1.10/24

{master:0}[edit]
```

7.4.4 Synchronizing the configuration to Virtual Chassis members

Whenever the configuration settings on the master switch are changed, the master switch must propagate changes to all other switches in the Virtual Chassis configuration. The CLI command in Example 7-32 illustrates how to synchronize the configuration.

Example 7-32 The commit synchronize command

```
{master:0}
ibm@J48E-VC> configure
Entering configuration mode
Users currently editing the configuration:
  root terminal u0 (pid 1731) on since 2010-02-14 19:04:07 UTC, idle 00:38:04
      {master:0}[edit]

{master:0}[edit]
ibm@J48E-VC# commit synchronize
fpc0:
configuration check succeeds
fpc1:
commit complete
fpc2:
commit complete
fpc0:
commit complete

{master:0}[edit]
```

You can also use the **commit synchronize** command with a **commit** command. You configure the command as shown in Example 7-33 and then enter the **commit** command.

Example 7-33 Alternative to the commit synchronize command

```
{master:0}[edit]
ibm@J48E-VC# set system commit synchronize

{master:0}[edit]

********
ibm@J48E-VC# commit
fpc0:
configuration check succeeds
fpc1:
commit complete
fpc2:
commit complete
fpc0:
commit complete

{master:0}[edit]
```

7.5 Operational monitoring of a Virtual Chassis

Commands are available that show information for all members in a Virtual Chassis configuration. These commands are helpful for determining the working condition of the Virtual Chassis configuration.

7.5.1 Verifying the member ID, status, and role of a Virtual Chassis member

To view member ID and role of individual members within the Virtual Chassis configuration, enter the CLI command shown in Example 7-34.

Example 7-34 Viewing the status of the Virtual Chassis

```
{master:0}
ibm@J48E-VC> show virtual-chassis status

Virtual Chassis ID: 6140.f8bf.28a0
                                          Mastership            Neighbor List
Member ID  Status   Serial No    Model    priority    Role      ID  Interface
0 (FPC 0)  Prsnt    BP0208174896 ex4200-48t      255  Master*    1  vcp-0
                                                                 2  vcp-1
1 (FPC 1)  Prsnt    BP0208110806 ex4200-48t      255  Backup     0  vcp-0
                                                                 2  vcp-1
2 (FPC 2)  Prsnt    134002W      ex4200-48t      128  Linecard   1  vcp-0
                                                                 0  vcp-1

Member ID for next new member: 3 (FPC 3)

{master:0}
```

The output shows three member switches in the Virtual Chassis configuration. The master switch has a mastership priority of 255, and its backup switch has the same mastership priority. It shows the interconnection of VCPs for each member switch.

7.5.2 Verifying the status of a Virtual Chassis port

To view the VCP status and associated details, enter the command shown in Example 7-35.

Example 7-35 Viewing the Virtual Chassis port status

```
{master:0}
ibm@J48E-VC> show virtual-chassis vc-port all-members
fpc0:
--------------------------------------------------------------------------
Interface   Type              Trunk  Status       Speed        Neighbor
or                             ID                 (mbps)       ID  Interface
PIC / Port
vcp-0       Dedicated           1    Up           32000        1   vcp-0
vcp-1       Dedicated           2    Up           32000        2   vcp-1

fpc1:
--------------------------------------------------------------------------
Interface   Type              Trunk  Status       Speed        Neighbor
or                             ID                 (mbps)       ID  Interface
PIC / Port
vcp-0       Dedicated           1    Up           32000        0   vcp-0
vcp-1       Dedicated           2    Up           32000        2   vcp-0

fpc2:
--------------------------------------------------------------------------
Interface   Type              Trunk  Status       Speed        Neighbor
or                             ID                 (mbps)       ID  Interface
PIC / Port
vcp-0       Dedicated           1    Up           32000        1   vcp-1
vcp-1       Dedicated           2    Up           32000        0   vcp-1

{master:0}
ibm@J48E-VC>
```

The output shows the status of VCP of each member switch and its peering connection to other member switch.

7.6 More information

For more information about Virtual Chassis, see the following resources:

- Virtual Chassis Configuration Guide

  http://www.juniper.net/techpubs/en_US/junos10.1/information-products/pathway-pages/ex-series/virtual-systems.html

- Virtual Chassis Technology Best Practices

  http://kb.juniper.net/index?page=content&id=TN31&cat=SWITCHING&actp=LIST

- Virtual Chassis installation

  https://www.juniper.net/us/en/training/elearning/ex_3200_4200.html

8

Layer 1: Configuration

This chapter explains how to configure Layer 1 capabilities of the various types of Ethernet interfaces on IBM j-type Ethernet switches and routers. It presents some usual configuration scenarios for data center implementations. For complete and advanced information, see the documentation listed in 8.4, “More information” on page 221.

This chapter includes the following topics:

- Network topology for Layer 1 configuration
- Configuration of the port settings
- Configuring PoE
- More information

Before reading this chapter, you must have a basic understanding of networking concepts. If necessary, see Chapter 1, “Fundamentals of Ethernet networking” on page 1, and *IBM j-type Data Center Networking Introduction*, SG24-7820, for a short introduction.

© Copyright IBM Corp. 2011. All rights reserved.

8.1 Network topology for Layer 1 configuration

Figure 8-1 illustrates the physical connections of the lab equipment that was used to demonstrate the examples in this chapter. Three units of IBM Ethernet Switch J48Es are interconnected to build a single management switch called the *Virtual Chassis switch*.

We used a generic wireless access point to demonstrate Power over Ethernet (PoE) capabilities of the IBM J48E switch. This access point was powered inline from the IBM Ethernet Switch J48E. Here only the connections that are relevant to this chapter are presented.

Figure 8-1 Network topology for the Layer 1 configuration

8.2 Configuration of the port settings

You can configure the speed, duplex, auto-negotiation, flow control, and other attributes of the IBM j-type Ethernet switches and routers physical ports as explained in the following sections.

8.2.1 Configuring the link settings on IBM j-type Ethernet switches

IBM j-type Ethernet switches include a factory default configuration that enables interfaces with the following link settings:

- All the Gigabit Ethernet interfaces are set to auto-negotiation.
- The speed for 10/100/1000 RJ45 (copper) Gigabit Ethernet interfaces is set to auto, so that the interface can operate at 10 Mbps, 100 Mbps, or 1 Gbps. The link operates at the highest possible speed, depending on the capabilities of the remote end. Also the ports are in auto-MDX mode.
- The flow control for Gigabit Ethernet interfaces and 10 Gigabit Ethernet interfaces is set to enabled.

- The link mode is set to auto, so that the interface can operate as either full duplex or half duplex. The link operates as full duplex unless this mode is not supported at the remote end.
- The 10 Gigabit Ethernet interfaces default to no auto-negotiation. The default speed is fixed to 10 Gbpsand the default link mode is fixed to full duplex.
- The Gigabit Ethernet fiber interfaces default to no auto-negotiation. The default speed is fixed to 1 Gbps and the default link mode is fixed to full duplex.

To change the default link parameters on Gigabit Ethernet interfaces for IBM j-type Ethernet switches, enter the following configuration level command:

```
set interface ge-fpc/pic/port ether-options [ speed [ auto-negotiation [ 10m-100m]
| 10m | 100m | 1g ] | link-mode [ automatic | full-duplex | half-duplex ] |
flow-control | no-flow-control | auto-negotiation | no-auto-negotiation ]
```

For a detailed explanation of the options for this command, see the *JUNOS Software System Basics and Services Command Reference* at the following web page:

http://www-01.ibm.com/support/docview.wss?uid=isg3T7000141&aid=1

To see the interface configuration, use the following command:

```
{master}[edit]
ibm@J08E-re0# show interface type-fpc/pic/port
```

After you commit the configuration by using the **commit** command, you can check the status of the interface by entering the following command:

```
{master}[edit]
ibm@J08E-re0> show interface type-fpc/pic/port
```

From a configuration level, you can use the **run** prefix:

```
{master}[edit]
ibm@J08E-re0# run show interface type-fpc/pic/port
```

Example 8-1 lists the output of the **show interface** command.

Example 8-1 The show interface operational level command

```
{master}
ibm@J08E-re0> show interfaces ge-1/0/0
Physical interface: ge-1/0/0, Enabled, Physical link is Up
  Interface index: 129, SNMP ifIndex: 245
  Link-level type: Ethernet, MTU: 1514, Speed: 100mbps, Duplex: Full-Duplex, BPDU
Error: None, MAC-REWRITE Error: None,
  Loopback: Disabled, Source filtering: Disabled, Flow control: Disabled,
Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:26:88:87:e2:31, Hardware address: 00:26:88:87:e2:31
  Last flapped   : 2010-05-05 14:34:21 UTC (00:44:51 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : None
  Active defects : None
```

```
  Logical interface ge-1/0/0.0 (Index 65) (SNMP ifIndex 501)
    Flags: SNMP-Traps 0x0 Encapsulation: ENET2
    Input packets : 0
    Output packets: 90
    Protocol inet
      Flags: None

{master}
ibm@J08E-re0> configure
Entering configuration mode

{master}[edit]
ibm@J08E-re0# run show interfaces ge-1/0/0
Physical interface: ge-1/0/0, Enabled, Physical link is Up
  Interface index: 129, SNMP ifIndex: 245
  Link-level type: Ethernet, MTU: 1514, Speed: 100mbps, Duplex: Full-Duplex, BPDU
Error: None, MAC-REWRITE Error: None,
  Loopback: Disabled, Source filtering: Disabled, Flow control: Disabled,
Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:26:88:87:e2:31, Hardware address: 00:26:88:87:e2:31
  Last flapped   : 2010-05-05 14:34:21 UTC (00:45:01 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 1768 bps (1 pps)
  Active alarms  : None
  Active defects : None

  Logical interface ge-1/0/0.0 (Index 65) (SNMP ifIndex 501)
    Flags: SNMP-Traps 0x0 Encapsulation: ENET2
    Input packets : 0
    Output packets: 91
    Protocol inlet
      Flags: None

{master}[edit]
ibm@J08E-re0#
```

If you must check the auto-negotiation status of a particular link, enter the following command:

```
ibm@J08E-re0> show interface type-fpc/pic/port media
```

Example 8-2 shows how to set the interface ge-1/0/0 of a J08E switch with a speed of 100 Mbps, full-duplex, and no flow control. It also shows how to verify the settings.

Example 8-2 Gigabit Ethernet options for IBM j-type Ethernet switches

```
{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/0 ether-options speed 100m

{master}[edit]
```

```
ibm@J08E-re0# set interfaces ge-1/0/0 ether-options link-mode full-duplex

{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/0 ether-options no-flow-control

{master}[edit]
ibm@J08E-re0# show interfaces ge-1/0/0
ether-options {
    no-flow-control;
    link-mode full-duplex;
    speed {
        100m;
    }
}
{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# run show interfaces ge-1/0/0 media
Physical interface: ge-1/0/0, Enabled, Physical link is Up
  Interface index: 129, SNMP ifIndex: 245
  Link-level type: Ethernet, MTU: 1514, Speed: 100mbps, Duplex: Full-Duplex, BPDU
Error: None, MAC-REWRITE Error: None, <---- speed and duplex configuration
  Loopback: Disabled, Source filtering: Disabled, Flow control: Disabled,
Auto-negotiation: Enabled,
  Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:26:88:87:e2:31, Hardware address: 00:26:88:87:e2:31
  Last flapped   : 2010-05-05 14:34:21 UTC (00:00:06 ago) <-- link flapped 6s ago
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : None
  Active defects : None
  MAC statistics:
    Input bytes: 431430816, Input packets: 550850, Output bytes: 431661792, Output
packets: 562085
  Autonegotiation information:
    Negotiation status: Complete <---- we still have link negotiation, but fixed
    Link partner:
        Link mode: Full-duplex, Flow control: Symmetric, Remote fault: OK, Link
partner Speed: 100 Mbps <------- Note the partner duplex, speed, and flow control
    Local resolution:
        Flow control: None, Remote fault: Link OK <--- Note the local flow control
```

```
{master}[edit]
ibm@J08E-re0#
```

If you change the speed and duplex of a link, the auto-negotiation of link parameters is still active to communicate the fixed values to the partner. To completely disable the auto-negotiation of link parameters, enter the following command:

```
{master}[edit]
ibm@J08E# set interfaces type-fpc/pic/port ether-options no-auto-negotiation
```

This command disables the link auto-negotiation as demonstrated in Example 8-3 where interface ge-1/0/0 is connected back to interface ge-1/0/16.

Example 8-3 Disabling link auto-negotiation

```
{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/0 ether-options no-auto-negotiation

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# run show interfaces ge-1/0/0 media
Physical interface: ge-1/0/0, Enabled, Physical link is Up
  Interface index: 129, SNMP ifIndex: 245
  Link-level type: Ethernet, MTU: 1514, Speed: 100mbps, Duplex: Full-Duplex, BPDU
Error: None,
  MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled, Flow
control: Disabled,
  Auto-negotiation: Disabled, Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:26:88:87:e2:31, Hardware address: 00:26:88:87:e2:31
  Last flapped   : 2010-05-05 15:57:36 UTC (00:13:52 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : None
  Active defects : None
  MAC statistics:
    Input bytes: 431443246, Input packets: 550906, Output bytes: 431704003, Output
packets: 562276
  Autonegotiation information:
    Negotiation status: No-autonegotiation <----- auto-negotiation is disabled

{master}[edit]
ibm@J08E-re0# run show interfaces ge-1/0/16 media
Physical interface: ge-1/0/16, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 261
```

```
  Link-level type: Ethernet, MTU: 1514, Speed: Auto, Duplex: Auto, BPDU Error:
None,
  MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled, Flow
control: Enabled,
  Auto-negotiation: Enabled, Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:26:88:87:e2:41, Hardware address: 00:26:88:87:e2:41
  Last flapped   : 2010-05-05 17:12:31 UTC (00:01:43 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : None
  Active defects : None
  MAC statistics:
    Input bytes: 663442, Input packets: 3002, Output bytes: 39070, Output packets:
176
  Autonegotiation information:
    Negotiation status: Incomplete <---- here the auto-negotiation is incomplete

{master}[edit]
ibm@J08E-re0#
```

Setting the mode on an SFP+ uplink module on J48E switch

SFP+ uplink modules are supported on J48E series switches. You can use these uplink modules either for two SFP+ transceivers or four SFP transceivers. You configure the operating mode on the module to match the type of transceiver you want to use. For SFP+ transceivers, you configure the 10 Gb operating mode, and for SFP transceivers, you configure the 1 Gb operating mode.

By default, the SFP+ uplink module operates in the 10 Gb mode and supports only SFP+ transceivers. If you have not changed the module from the default setting and you want to use SFP+ transceivers, you are not required to configure the operating mode.

To set the operating mode of an SFP+ uplink module, change the operating mode to the appropriate mode for the transceiver type that you want to use by using one of the following configuration level commands:

```
[edit]
ibm@J48E# set chassis fpc 0 pic 1 sfpplus pic-mode 1g

[edit]
ibm@J48E# set chassis fpc 0 pic 1 sfpplus pic-mode 10g
```

If the switch is running Junos Release 10.1 or later, the changed operating mode takes effect immediately unless a port on the SFP+ uplink module is a Virtual Chassis port (VCP). If any port on the SFP+ uplink module is a VCP, the changed operating mode does not take effect until the next reboot of the switch.

Packet Forwarding Engine: During the operating mode change, the Packet Forwarding Engine is restarted. In a Virtual Chassis configuration, this restart means that the Flexible PIC Concentrator connection with the master is dropped and then reconnected.

If the switch is running Junos Release 10.0 or earlier, reboot the switch.

You can see whether the operating mode has been changed to the new mode that you configured by entering the following operational level command:

```
show chassis pic fpc-slot slot-number pic-slot 1
```

8.2.2 Configuring the link settings on IBM j-type Ethernet routers

This section lists the commands that are used to change the default link parameters on 10 Gb and Gigabit Ethernet interfaces for IBM j-type Ethernet routers.

For no auto-negotiation of link parameters, use the following command:

```
[edit]
ibm@J11M# set interfaces type-fpc/pic/port gigether-options no-auto-negotiation
```

For auto-negotiation of link parameters, use the following command:

```
[edit]
ibm@J11M# set interfaces type-fpc/pic/port gigether-options auto-negotiation
```

To disable flow control on the link, use the following command:

```
[edit]
ibm@J11M# set interfaces type-fpc/pic/port gigether-options no-flow-control
```

To enable flow control on the link, use the following command:

```
[edit]
ibm@J11M# set interfaces type-fpc/pic/port gigether-options flow-control
```

To change the speed on 10/100/1000 copper interfaces, use the following command:

```
[edit]
ibm@J11M# set interfaces ge-fpc/pic/port speed [ 10m | 100m | 1g ]
```

For fixed full-duplex capability on 10/100/1000 copper interfaces, use the following command:

```
[edit]
ibm@J11M# set interfaces ge-fpc/pic/port link-mode full-duplex
```

Example 8-4 demonstrates how to set the interface ge-2/0/0 of a J11M router with a speed of 100 Mbps, full-duplex, and no flow-control.

Example 8-4 Gigabit Ethernet options for IBM j-type Ethernet routers

```
[edit]
ibm@J11M-re0# set interfaces ge-2/0/0 speed 100m

[edit]
ibm@J11M-re0# set interfaces ge-2/0/0 link-mode full-duplex

[edit]
ibm@J11M-re0# set interfaces ge-2/0/0 gigether-options no-flow-control

[edit]
ibm@J11M-re0# show interfaces ge-2/0/0
speed 100m;
link-mode full-duplex;
gigether-options {
```

```
    no-flow-control;
}

[edit]
ibm@J11M-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

[edit]
```

8.2.3 Common configuration options for interfaces

This section demonstrates how to configure several common options on IBM j-type Ethernet switches and routers.

Administratively disabling an interface

To disable an interface, use this configuration mode command:

```
[edit]
ibm@J11M# set interfaces type-fpc/pic/port disable
```

Configuring an interface description

As a good practice, have an interface description for documentation and troubleshooting purposes. Add a description to an interface by using this command:

```
[edit]
ibm@J11M# set interfaces type-fpc/pic/port description interface_description
```

Configuring a Maximum Transmit Unit

To change the Maximum Transmit Unit (MTU) of an interface, use this configuration mode command:

```
{master}[edit]
ibm@J08E# set interfaces type-fpc/pic/port mtu mtu_value_between_256..9216
```

Example 8-5 shows how to set the description of interface xe-3/0/0 to "`Link to backbone`", set its MTU value to 4096, and disable the interface ge-1/0/0.

Example 8-5 Other interface options

```
{master}[edit]
ibm@J08E-re0# set interfaces xe-3/0/0 description "Link to backbone"

{master}[edit]
ibm@J08E-re0# set interfaces xe-3/0/0 mtu 4096

{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/0 disable

{master}[edit]
ibm@J08E-re0# commit
```

```
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# run show interfaces xe-3/0/0 descriptions
Interface       Admin Link Description
xe-3/0/0        up    down Link to backbone

{master}[edit]
ibm@J08E-re0# run show interfaces xe-3/0/0
Physical interface: xe-3/0/0, Enabled, Physical link is Down
  Interface index: 225, SNMP ifIndex: 140
  Description: Link to backbone
  Link-level type: Ethernet, MTU: 4096, Speed: 10Gbps, Duplex: Full-Duplex, BPDU
Error: None, <--------- note the MTU size on this line
  MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled, Flow
control: Enabled
  Device flags   : Present Running
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:26:88:87:e2:91, Hardware address: 00:26:88:87:e2:91
  Last flapped   : Never
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : LINK
  Active defects : LINK

{master}[edit]
ibm@J08E-re0# run show interfaces ge-1/0/0
Physical interface: ge-1/0/0, Administratively down, Physical link is Down
  Interface index: 129, SNMP ifIndex: 245 <---- note the “Administratively down”
  Link-level type: Ethernet, MTU: 1514, Speed: 100mbps, Duplex: Full-Duplex, BPDU
Error: None,
  MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled, Flow
control: Disabled,
  Auto-negotiation: Disabled, Remote fault: Online
  Device flags   : Present Running Down
  Interface flags: Hardware-Down Down SNMP-Traps Internal: 0x0
  Link flags     : None
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:26:88:87:e2:31, Hardware address: 00:26:88:87:e2:31
  Last flapped   : 2010-05-05 19:42:10 UTC (00:03:03 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)
  Active alarms  : LINK
  Active defects : LINK

  Logical interface ge-1/0/0.0 (Index 65) (SNMP ifIndex 501)
    Flags: Device-Down SNMP-Traps 0x0 Encapsulation: ENET2
    Input packets : 472
```

```
    Output packets: 609
    Protocol inet
      Flags: None

{master}[edit]
```

8.3 Configuring PoE

From the IBM j-type Ethernet switches and routers, only the J48E switch supports PoE and only on the first 8 ports.

8.3.1 Configuring PoE on an IBM Ethernet Switch J48E

For a J48E switch, the factory default configuration specifies and enables PoE interfaces for the PoE ports.

To disable PoE on all interfaces of a J48E switch, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe interface all disable
```

To disable PoE on a specific interface of a J48E switch, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe interface type-fpc/pic/port disable
```

To enable PoE on all interfaces of a J48E switch, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe interface all
```

To enable PoE on a specific interface of an IBM J48E, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe interface type-fpc/pic/port
```

To enable PoE on all interfaces that were previously disabled, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# delete poe interface all disable
```

To enable PoE on a specific interface that was previously disabled, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# delete poe interface type-fpc/pic/port disable
```

By default, the power management mode is static. To change the power management mode to *class*, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe management class
```

To change the power management mode back to default static mode, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe management static
```

To set the power priority for all interfaces, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe interface all priority [ low | high ]
```

To set the power priority for all interfaces, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe interface type-fpc/pic/port priority [ low | high ]
```

Default wattage available for all interfaces is 15.4 W.

To set the maximum PoE wattage available for all interfaces, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe interface all maximum-power max_power_between_0..15.4
```

To set the maximum PoE wattage that is available for a specific interface, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe interface type-fpc/pic/port maximum-power
max_power_between_0..15.4
```

To enable logging of PoE power consumption with the default telemetries settings for all interfaces, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe interface all telemetries
```

To enable logging of PoE power consumption with the default telemetries settings for a specific interface, use the following command:

```
{master:0}[edit]
ibm@J48E-VC# set poe interface type-fpc/pic/port telemetries
```

To reserve a specified wattage of power for the switch during a spike in PoE consumption, use the following command. The default is 0.

```
{master:0}[edit]
ibm@J48E-VC# set poe guard-band guard_band_value_between_0..19_watts
```

Example 8-6 shows the PoE configuration from three J48E switches bundled in a virtual chassis configuration.

Example 8-6 PoE configuration

```
{master:0}[edit]
ibm@J48E-VC# set poe interface all

{master:0}[edit]
ibm@J48E-VC# set poe interface ge-0/0/0 disable

{master:0}[edit]
ibm@J48E-VC# set poe interface all telemetries interval 1 duration 24

{master:0}[edit]
ibm@J48E-VC# set poe management class

{master:0}[edit]
ibm@J48E-VC# set poe guard-band 15
```

```
{master:0}[edit]
ibm@J48E-VC# set poe interface ge-0/0/1 telemetries

{master:0}[edit]
ibm@J48E-VC# set poe interface ge-0/0/1 priority high

{master:0}[edit]
ibm@J48E-VC# set poe interface all priority low

{master:0}[edit]
ibm@J48E-VC# set poe interface ge-0/0/1 maximum-power 7

{master:0}[edit]
ibm@J48E-VC# set poe interface all maximum-power 15.4

{master:0}[edit]
ibm@J48E-VC# show poe
management class;
guard-band 15;
interface all {
    priority low;
    maximum-power 15.4;
    telemetries {
        interval 1;
        duration 24;
    }
}
interface ge-0/0/0 {
    disable;
}
interface ge-0/0/1 {
    priority high;
    maximum-power 7;
    telemetries;
}

{master:0}[edit]
ibm@J48E-VC# commit
fpc0:
configuration check succeeds
fpc1:
commit complete
fpc2:
commit complete
fpc0:
commit complete

{master:0}[edit]
ibm@J48E-VC#
```

8.3.2 Verifying the status of PoE interfaces on an IBM Ethernet Switch J48E

Verify that the PoE interfaces are enabled and set to the desired priority settings by entering the following command:

```
{master:0}
ibm@J48E-VC> show poe interface [type-fpc/pic/port]
```

To view the status of the PoE controller, use the following command:

```
{master:0}
ibm@J48E-VC> show poe controller
```

To view a history of power consumption on the specified interface, use the following command:

```
{master:0}
ibm@J48E-VC> show poe telemetries interface type-fpc/pic/port all
```

Example 8-7 shows the output of the PoE verification commands. A PoE device is attached on port ge-2/0/0.

Example 8-7 PoE verification commands output

```
{master:0}
ibm@J48E-VC> show poe interface
Interface Admin status Oper status Max power Priority Power consumption Class
 ge-0/0/0 Enabled      OFF         15.4W     Low      0.0W                0
 ge-0/0/1 Enabled      OFF         15.4W     Low      0.0W                0
 ge-0/0/2 Enabled      OFF         15.4W     Low      0.0W                0
 ge-0/0/3 Enabled      OFF         15.4W     Low      0.0W                0
 ge-0/0/4 Enabled      OFF         15.4W     Low      0.0W                0
 ge-0/0/5 Enabled      OFF         15.4W     Low      0.0W                0
 ge-0/0/6 Enabled      OFF         15.4W     Low      0.0W                0
 ge-0/0/7 Enabled      OFF         15.4W     Low      0.0W                0
 ge-1/0/0 Enabled      OFF         15.4W     Low      0.0W                0
 ge-1/0/1 Enabled      OFF         15.4W     Low      0.0W                0
 ge-1/0/2 Enabled      OFF         15.4W     Low      0.0W                0
 ge-1/0/3 Enabled      OFF         15.4W     Low      0.0W                0
 ge-1/0/4 Enabled      OFF         15.4W     Low      0.0W                0
 ge-1/0/5 Enabled      OFF         15.4W     Low      0.0W                0
 ge-1/0/6 Enabled      OFF         15.4W     Low      0.0W                0
 ge-1/0/7 Enabled      OFF         15.4W     Low      0.0W                0
 ge-2/0/0 Enabled       ON         15.4W     Low      3.8W                3 <- PoE
 ge-2/0/1 Enabled      OFF         15.4W     Low      0.0W                0
 ge-2/0/2 Enabled      OFF         15.4W     Low      0.0W                0
 ge-2/0/3 Enabled      OFF         15.4W     Low      0.0W                0
 ge-2/0/4 Enabled      OFF         15.4W     Low      0.0W                0
 ge-2/0/5 Enabled      OFF         15.4W     Low      0.0W                0
 ge-2/0/6 Enabled      OFF         15.4W     Low      0.0W                0
 ge-2/0/7 Enabled      OFF         15.4W     Low      0.0W                0

{master:0}
ibm@J48E-VC> show poe controller
Controller  Maximum   Power              Guard band  Management
index       power     consumption
   0        130 W       0W                 0W        Static
   1        130 W       0W                 0W        Static
```

```
   2        130 W       3W                 0W        Static <--- 3W consumption

{master:0}
ibm@J48E-VC> show poe telemetries interface ge-2/0/0 all
Sl No    Timestamp                Power    Voltage
   1     02-20-2010 21:10:33 UTC  3.8W     50.5V
   2     02-20-2010 21:09:33 UTC  3.8W     50.5V
   3     02-20-2010 21:08:33 UTC  0.0W     50.6V
   4     02-20-2010 21:07:33 UTC  0.0W     50.6V
   5     02-20-2010 21:06:33 UTC  0.0W     50.6V

{master:0}
ibm@J48E-VC> show poe interface ge-2/0/0
PoE interface status:
PoE interface                :  ge-2/0/0
Administrative status        : Enabled
Operational status           :   ON
Power limit on the interface : 15.4W
Priority                     : Low
Power consumed               : 3.7W
Class of power device        :   3

{master:0}
ibm@J48E-VC>
```

8.4 More information

You can find all of the following documents on the IBM support site at this address:

http://www.ibm.com/systems/support/supportsite.wss/brandmain?brandind=5375876

For additional information and step-by-step installation procedures, see the following documentation that is appropriate for your switch:

- *IBM Ethernet Switch J48E Complete Hardware Guide,* GA32-0663
- *IBM Ethernet Switch J48E Quick Start,* GA32-0664
- *IBM Ethernet Switch J08E Complete Hardware Guide,* GA32-0665
- *IBM Ethernet Switch J08E Quick Start,* GA32-0666
- *IBM Ethernet Switch J16E Complete Hardware Guide,* GA32-0667
- *IBM Ethernet Switch J16E Quick Start,* GA32-0668

For more information about the IBM j-type Ethernet routers the related hardware, see the following documentation:

- *IBM j-type m-series Ethernet Router Dense Port Concentrators (DPC) Guide,* GA32-0661
- *IBM j-type m-series Ethernet Routing Engine Installation Instructions,* GA32-0682
- *IBM j-type m-series Ethernet Services PIC Guide,* GA32-0662
- *IBM Ethernet Router J02M Hardware Guide,* GA32-0655
- *IBM Ethernet Router J02M Quick Start,* GA32-0656
- *IBM Ethernet Router J06M Hardware Guide,* GA32-0657

- *IBM Ethernet Router J06M Quick Start,* GA32-0658
- *IBM Ethernet Router J11M Hardware Guide,* GA32-0659
- *IBM Ethernet Router J11M Quick Start,* GA32-0660
- *JUNOS Software IBM j-type m-series Ethernet Routers Solutions Guide,* GA32-0683

For more information about Junos software, see the following documentation:

- *Juniper Web Device Manager for IBM j-type Ethernet Switches and Routers Interface User Guide,* GA32-0688
- *JUNOS Software Access Privilege Configuration Guide,* GA32-0696
- *JUNOS Software Broadband Subscriber Management Solutions Guide,* GA32-0709
- *JUNOS Software Class of Service Configuration Guide,* GA32-0738
- *JUNOS Software CLI User Guide,* GA32-0697
- *JUNOS Software Configuration and Diagnostic Automation Guide,* GA32-0679
- *JUNOS Software Ethernet Routing Engine Media Upgrade Kit,* GA32-0681
- *JUNOS Software Feature Guide,* GA32-0739
- *JUNOS Software Hierarchy and RFC Reference,* GA32-0712
- *JUNOS Software High Availability Configuration Guide,* GA32-0670
- *JUNOS Software IBM j-type m-series Ethernet Routers Layer 2 Configuration Guide,* GA32-0708
- *JUNOS Software Installation and Upgrade Guide,* GA32-0695
- *JUNOS Software Interfaces Command Reference,* GA32-0672
- *JUNOS Software JUNOScript API Guide,* GA32-0674
- *JUNOS Software MPLS Applications Configuration Guide,* GA32-0702
- *JUNOS Software Multicast Protocols Configuration Guide,* GA32-0703
- *JUNOS Software NETCONF API Guide,* GA32-0678
- *JUNOS Software Network Interfaces Configuration Guide,* GA32-0706
- *JUNOS Software Network Management Configuration Guide,* GA32-0698
- *JUNOS Software Policy Framework Configuration Guide,* GA32-0704
- *JUNOS Software Routing Protocols and Policies Command Reference,* GA32-0673
- *JUNOS Software Services Interfaces Configuration Guide,* GA32-0707
- *JUNOS Software Subscriber Access Configuration Guide,* GA32-0711
- *JUNOS Software System Basics and Services Command Reference,* GA32-0671
- *JUNOS Software System Log Messages Reference,* GA32-0675
- *JUNOS Software VPNs Configuration Guide,* GA32-0705
- *JUNOScope Software User Guide,* GA32-0670

9

Layer 2: Switching configuration

This chapter explains how to configure Layer 2 capabilities on IBM j-type Ethernet switches and routers. It presents some usual configuration scenarios for data center implementations. For complete and advanced information, see the additional documents listed in 9.7, “More information” on page 270.

This chapter includes the following topics:

- Network topology for Layer 2 configuration
- Link aggregation configuration
- Configuring a VLAN
- Configuring a port VLAN
- Configuring the Spanning Tree Protocol
- Link Layer Discovery Protocol
- More information

Before reading this chapter, you must have a basic understanding of networking concepts. If necessary, see Chapter 1, “Fundamentals of Ethernet networking” on page 1, and *IBM j-type Data Center Networking Introduction*, SG24-7820, for a short introduction.

© Copyright IBM Corp. 2011. All rights reserved.

9.1 Network topology for Layer 2 configuration

Figure 9-1 illustrates the physical connections of the lab equipment used for demonstrating the examples from this chapter. It shows only the connections that are relevant to this chapter. Three units of IBM Ethernet Switch J48Es are interconnected to build a single management switch called the *Virtual Chassis switch*.

Figure 9-1 Network topology for Layer 2 configurations

9.2 Link aggregation configuration

This section provides details about link aggregation implementation in IBM j-type Ethernet switches and routers. It also explains how to configure and verify link aggregation configuration in Junos.

9.2.1 Link aggregation group platform characteristics

You configure a link aggregation group (LAG) by specifying the link number as a physical device and then associate a set of interfaces (ports) with the link. All the interfaces must have the same speed and be in full-duplex mode. Junos for IBM j-type Ethernet switches assigns a unique ID and port priority to each interface. The ID and priority are not configurable.

The number of interfaces that can be grouped into a LAG and the total number of LAGs supported on a switch vary by switch model. Table 9-1 lists the maximum number of interfaces per LAG and the maximum number of LAGs they support for the IBM j-type Ethernet switches.

Table 9-1 Maximum interfaces per LAG and maximum LAGs per equipment

Equipment model	Maximum interfaces per LAG	Maximum LAGs
J48E	8	64
J08E and J16E	12	255
J11M, J06M, and J02M	16	480

When configuring LAGs, consider the following guidelines and considerations:

- Configure the LAG on both sides of the link.
- Set the interfaces on either side of the link to the same speed.
- Configure and apply firewall filters on a LAG.
- Optional: Configure the Link Aggregation Control Protocol (LACP) for link negotiation and misconfiguration detection.
- Member links are not required to be contiguous ports.
- On IBM j-type Ethernet switches, the hashing algorithm is fixed and cannot be changed.
- On IBM j-type Ethernet routers, the hashing algorithm can be changed.
- Processor control packets are always sent on the lowest member link.

Physical Ethernet ports: You can combine physical Ethernet ports belonging to different member switches of a Virtual Chassis configuration to form a LAG.

9.2.2 Configuring link aggregation

In Junos, the name for link aggregation interfaces is `ag`*x*, where *x* refers to the instance number of the aggregation interface and has local significance. As a good practice, you must have the same interface number on both heads of a LAG.

Configuring a LAG on IBM j-type Ethernet switches and routers entails the following tasks:

- Configuring the maximum number of aggregated interfaces
- Configuring the aggregated interface parameters
- Configuring a protocol family for the aggregated interface
- Configuring the member interfaces that form the LAG

Configuring the maximum number of aggregated interfaces

To configure the maximum number of aggregated interfaces, use the following command:

```
{master}[edit]
ibm@J08E-re0# set chassis aggregated-devices ethernet device-count x
```

where *x* is the device count between 1 and the maximum number of LAGs supported by the platform.

Configuring the aggregated interface parameters

To configure the minimum links for the aggregated interface, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces aex aggregated-ether-options minimum-links y
```

where:

x is the LAG number.

y is the number of interfaces in this LAG with values between 1 and the maximum number of interfaces per LAG supported by the platform.

Configuring a protocol family for the aggregated interface

For a LAG to be functional, at least one protocol family must be defined on a logical unit. Without a defined protocol, nothing can be transported over the LAG. The interface will not be active. Many protocols can be active on the LAG. The following example shows how to configure the **inet** and **ethernet-switching** protocol families because they are the most common in data center scenarios. The **inet** protocol is available for IBM j-type Ethernet switches and routers. The **ethernet-switching** protocol is available only for IBM j-type Ethernet switches.

To configure the **inet** protocol family on a LAG in IBM j-type Ethernet switches and routers, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces aex unit 0 family inet
```

To configure the **ethernet-switching** protocol family on a LAG in IBM j-type Ethernet switches, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces aex unit 0 family ethnernet-switching
```

where `aex` is the defined LAG interface.

Configuring the member interfaces that form the LAG

To configure IBM j-type Ethernet switches on the member interfaces that take part in the LAG, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces type-fpc/pic/port ether-options 802.3ad aex
```

where `aex` is the defined LAG interface.

To configure on IBM j-type Ethernet routers the member interfaces that take part in the LAG, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces type-fpc/pic/port gigether-options 802.3ad aex
```

where aex is the defined LAG interface.

9.2.3 Verifying link aggregation

To verify the operation of a LAG, you can use several **show** commands. To verify the LAG interface status, use the following command:

```
{master}
ibm@J08E-re0> show interfaces aex extensive
```

where aex is the defined LAG interface.

To verify the LACP interface parameters and negotiation status, use the following command:

```
{master}
ibm@J08E-re0> show lacp interfaces [aex]
```

where aex is a defined LAG interface and is an optional argument. If aex is not specified, all LAG interfaces are listed.

To view the LACP statistics, use the following command:

```
{master}
ibm@J08E-re0> show lacp statistics interfaces [aex]
```

where aex is a defined LAG interface and is an optional argument. If aex is not specified, statistics for all LAG interfaces are listed.

To clear the LACP statistics on all or specific interface, use the following command:

```
{master}
ibm@J08E-re0> clear lacp statistics interfaces [aex]
```

where aex is a defined LAG interface and is an optional argument. If aex is not specified, statistics for all LAG interfaces are cleared.

9.2.4 A sample LAG configuration

Example 9-1 shows the process of creating and verifying a LAG between a J08E switch and a J48E switch. The LAG has four member interfaces with a minimum of three active interfaces to be functional.

Example 9-1 A static LAG configuration

```
{master}[edit]
ibm@J08E-re0# set chassis aggregated-devices ethernet device-count 1

{master}[edit]
ibm@J08E-re0# set interfaces ae0 aggregated-ether-options minimum-links 3

{master}[edit]
ibm@J08E-re0# set interfaces ae0 unit 0 family ethernet-switching
```

```
{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/1 ether-options 802.3ad ae0

{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/2 ether-options 802.3ad ae0

{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/3 ether-options 802.3ad ae0

{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/4 ether-options 802.3ad ae0

{master}[edit]
ibm@J08E-re0# show interfaces ae0
aggregated-ether-options {
    minimum-links 3;
}
unit 0 {
    family ethernet-switching;
}

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# exit
Exiting configuration mode

{master}
ibm@J08E-re0> show interfaces ae0 extensive
Physical interface: ae0, Enabled, Physical link is Up <------ link is up
  Interface index: 234, SNMP ifIndex: 508, Generation: 237
  Link-level type: Ethernet, MTU: 1514, Speed: 4Gbps, BPDU Error: None,
MAC-REWRITE Error: None, <------------------ note the LAG speed of 4 Gbps
  Loopback: Disabled, Source filtering: Disabled, Flow control: Disabled, Minimum
links needed: 3,
  Minimum bandwidth needed: 0
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Current address: 00:26:88:87:e2:01, Hardware address: 00:26:88:87:e2:01
  Last flapped   : 2010-05-06 19:24:36 UTC (00:38:55 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :           1611836522                    0 bps
   Output bytes  :           1307667066                    0 bps
   Input  packets:              2089876                    0 pps
   Output packets:               959763                    0 pps
   IPv6 transit statistics:
    Input  bytes  :                  0
```

```
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
0, Resource errors: 0
  Output errors:
    Carrier transitions: 234, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
0

  Logical interface ae0.0 (Index 65) (SNMP ifIndex 509) (Generation 138)
    Flags: SNMP-Traps 0x0 Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :             0          0             0            0
        Output:           992          0        102005            0
    Link:
      ge-1/0/1.0
        Input :             0          0             0            0
        Output:            13          0          2192            0
      ge-1/0/2.0
        Input :             0          0             0            0
        Output:          1072          0        120878            0
      ge-1/0/3.0
        Input :             0          0             0            0
        Output:           115          0         25841            0
      ge-1/0/4.0
        Input :             0          0             0            0
        Output:           115          0         25841            0
    Marker Statistics:   Marker Rx     Resp Tx   Unknown Rx   Illegal Rx
      ge-1/0/1.0                0           0            0            0
      ge-1/0/2.0                0           0            0            0
      ge-1/0/3.0                0           0            0            0
      ge-1/0/4.0                0           0            0            0
    Protocol eth-switch, Generation: 157, Route table: 0
      Flags: None

{master}
ibm@J08E-re0> show lacp interfaces ae0
Aggregated interface: ae0 <--- LACP not active on ae0, no information, and no
statistics

{master}
ibm@J08E-re0> show lacp statistics interfaces ae0

{master}
ibm@J08E-re0>
```

Example 9-1 on page 227 shows that a LAG with the name ae0 that has four active gigabit member links. No information is available about the LACP and the LACP statistics because the LACP protocol is not enabled on the LAG.

By using the same configuration of an `ae0` LAG between the J08E switch and the J48E switch, we investigate what happens if we lose some of the four member links. Example 9-2 shows verifying the status of the `ae0` LAG with two active links and then three active links. To avoid a physical unplugging of the cable, the relevant interfaces are disabled.

Example 9-2 Minimum links in a LAG

```
{master}
ibm@J08E-re0> configure
Entering configuration mode

{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/1 disable <--------- shut down this interface

{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/2 disable <--------- shut down this interface

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# run show interfaces ae0 extensive
Physical interface: ae0, Enabled, Physical link is Down <---the interface is down
  Interface index: 234, SNMP ifIndex: 508, Generation: 237
  Link-level type: Ethernet, MTU: 1514, Speed: Unspecified, BPDU Error: None,
MAC-REWRITE Error: None,
  Loopback: Disabled, Source filtering: Disabled, Flow control: Disabled, Minimum
links needed: 3, <------------------------- minimum of links 3, but we only have 2
  Minimum bandwidth needed: 0
  Device flags   : Present Running
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
  Current address: 00:26:88:87:e2:01, Hardware address: 00:26:88:87:e2:01
  Last flapped   : 2010-05-06 20:20:30 UTC (00:00:21 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :          1611836522                    0 bps
   Output bytes  :          1307667066                    0 bps
   Input  packets:             2089876                    0 pps
   Output packets:              959763                    0 pps
   IPv6 transit statistics:
    Input  bytes  :                  0
    Output bytes  :                  0
    Input  packets:                  0
    Output packets:                  0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
0, Resource errors: 0
  Output errors:
    Carrier transitions: 234, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
0
```

```
  Logical interface ae0.0 (Index 65) (SNMP ifIndex 509) (Generation 138)
    Flags: Hardware-Down Device-Down SNMP-Traps 0x0 Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :             0          0             0            0
        Output:          1502          0        154478            0
    Link:
      ge-1/0/1.0 <-- down
        Input :             0          0             0            0
        Output:           556          0         62813            0
      ge-1/0/2.0 <-- down
        Input :             0          0             0            0
        Output:          1107          0        129118            0
      ge-1/0/3.0
        Input :             0          0             0            0
        Output:           150          0         34276            0
      ge-1/0/4.0
        Input :             0          0             0            0
        Output:           150          0         34276            0
    Marker Statistics:   Marker Rx     Resp Tx   Unknown Rx   Illegal Rx
      ge-1/0/1.0                0           0            0            0
      ge-1/0/2.0                0           0            0            0
      ge-1/0/3.0                0           0            0            0
      ge-1/0/4.0                0           0            0            0
    Protocol eth-switch, Generation: 157, Route table: 0
      Flags: None

{master}[edit]
ibm@J08E-re0# delete interfaces ge-1/0/2 disable <--- enable the interface again

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# run show interfaces ae0 extensive
Physical interface: ae0, Enabled, Physical link is Up <---- the interface is up
  Interface index: 234, SNMP ifIndex: 508, Generation: 237
  Link-level type: Ethernet, MTU: 1514, Speed: 3Gbps, BPDU Error: None,
MAC-REWRITE Error: None, <--------------------------------- link speed is 3 Gpbs
  Loopback: Disabled, Source filtering: Disabled, Flow control: Disabled, Minimum
links needed: 3,
  Minimum bandwidth needed: 0
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Current address: 00:26:88:87:e2:01, Hardware address: 00:26:88:87:e2:01
  Last flapped   : 2010-05-06 20:27:58 UTC (00:00:04 ago)
  Statistics last cleared: Never
  Traffic statistics:
```

```
   Input  bytes  :           1611875532                    0 bps
   Output bytes  :           1307740632                  512 bps
   Input  packets:              2090042                    0 pps
   Output packets:               960443                    1 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
0, Resource errors: 0
  Output errors:
    Carrier transitions: 237, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
0

  Logical interface ae0.0 (Index 65) (SNMP ifIndex 509) (Generation 138)
    Flags: SNMP-Traps 0x0 Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :             0          0             0            0
        Output:          1506          0        154833            0
    Link:
      ge-1/0/1.0 <-- down
        Input :             0          0             0            0
        Output:           556          0         62813            0
      ge-1/0/2.0
        Input :             0          0             0            0
        Output:          1111          0        129473            0
      ge-1/0/3.0
        Input :             0          0             0            0
        Output:           164          0         37650            0
      ge-1/0/4.0
        Input :             0          0             0            0
        Output:           164          0         37650            0
    Marker Statistics:   Marker Rx     Resp Tx   Unknown Rx   Illegal Rx
      ge-1/0/1.0                0           0            0            0
      ge-1/0/2.0                0           0            0            0
      ge-1/0/3.0                0           0            0            0
      ge-1/0/4.0                0           0            0            0
    Protocol eth-switch, Generation: 157, Route table: 0
      Flags: None
```

9.2.5 Configuring link aggregation with LACP

So far, only statically defined LAG interfaces have been configured. For link aggregation negotiation and misconfiguration detection, we configure LACP for LAG interface on both sides of the link.

To configure LACP active mode, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces aex aggregated-ether-options lacp active
```

where `aex` is the defined LAG interface.

To configure LACP passive mode, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces aex aggregated-ether-options lacp passive
```

where aex is the defined LAG interface.

9.2.6 Sample LAG with LACP configuration

By using the same configuration of an ae0 LAG between the J08E switch and the J48E switch, we configure LACP only on one side of the link, on J08E switch. Example 9-2 on page 230 shows a LAG with four member links with one of the links disabled. Example 9-3 shows how to configure ae0 as an active LACP speaker and how to verify the results.

Example 9-3 Configuring an LACP-enabled LAG on only one side of the LAG

```
{master}[edit]
ibm@J08E-re0# set interfaces ae0 aggregated-ether-options lacp active

{master}[edit]
ibm@J08E-re0# show interfaces ae0
aggregated-ether-options {
    minimum-links 3;
    lacp {
        active;
    }
}
unit 0 {
    family ethernet-switching;
}

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# run show interfaces ae0 extensive
Physical interface: ae0, Enabled, Physical link is Down <---- LAG port is down
  Interface index: 234, SNMP ifIndex: 508, Generation: 237
  Link-level type: Ethernet, MTU: 1514, Speed: Unspecified, BPDU Error: None,
MAC-REWRITE Error: None,
  Loopback: Disabled, Source filtering: Disabled, Flow control: Disabled, Minimum
links needed: 3,
  Minimum bandwidth needed: 0
  Device flags   : Present Running
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x0
  Current address: 00:26:88:87:e2:01, Hardware address: 00:26:88:87:e2:01
  Last flapped   : 2010-05-06 21:24:21 UTC (00:02:11 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :           1611897152                    0 bps
```

```
   Output bytes  :           1307792228                    0 bps
   Input  packets:              2090134                    0 pps
   Output packets:               960989                    0 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
0, Resource errors: 0
  Output errors:
    Carrier transitions: 237, Errors: 0, Drops: 0, MTU errors: 0, Resource errors:
0

  Logical interface ae0.0 (Index 65) (SNMP ifIndex 509) (Generation 138)
    Flags: Hardware-Down Device-Down SNMP-Traps 0x0 Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :             0          0             0            0
        Output:          3194          0        328697            0
    Link:
      ge-1/0/1.0 <-- down
        Input :             0          0             0            0
        Output:           556          0         62813            0
      ge-1/0/2.0
        Input :             0          0             0            0
        Output:          2916          0        331534            0
      ge-1/0/3.0
        Input :             0          0             0            0
        Output:           281          0         65847            0
      ge-1/0/4.0
        Input :             0          0             0            0
        Output:           281          0         65847            0
    LACP info:        Role     System             System      Port    Port  Port
                            priority          identifier  priority  number   key
      ge-1/0/1.0     Actor          0  00:26:88:87:e3:f0       127     577     1
      ge-1/0/1.0   Partner          0  00:00:00:00:00:00         0       0     0
      ge-1/0/2.0     Actor          0  00:26:88:87:e3:f0       127     578     1
      ge-1/0/2.0   Partner          0  00:00:00:00:00:00         0       0     0
      ge-1/0/3.0     Actor          0  00:26:88:87:e3:f0       127     579     1
      ge-1/0/3.0   Partner          0  00:00:00:00:00:00         0       0     0
      ge-1/0/4.0     Actor          0  00:26:88:87:e3:f0       127     580     1
      ge-1/0/4.0   Partner          0  00:00:00:00:00:00         0       0     0
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      ge-1/0/1.0                 0           0            0            0
      ge-1/0/2.0                 0         131            0            0
      ge-1/0/3.0                 0         131            0            0
      ge-1/0/4.0                 0         131            0            0
    Marker Statistics:   Marker Rx     Resp Tx   Unknown Rx   Illegal Rx
      ge-1/0/1.0                 0           0            0            0
      ge-1/0/2.0                 0           0            0            0
      ge-1/0/3.0                 0           0            0            0
      ge-1/0/4.0                 0           0            0            0
    Protocol eth-switch, Generation: 157, Route table: 0
```

```
      Flags: None

{master}[edit]
ibm@J08E-re0# run show lacp interfaces
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity
      ge-1/0/1       Actor    No   Yes    No   No   No   Yes     Fast    Active
      ge-1/0/1     Partner    No   Yes    No   No   No   Yes     Fast   Passive
      ge-1/0/2       Actor    No   Yes    No   No   No   Yes     Fast    Active
      ge-1/0/2     Partner    No   Yes    No   No   No   Yes     Fast   Passive
      ge-1/0/3       Actor    No   Yes    No   No   No   Yes     Fast    Active
      ge-1/0/3     Partner    No   Yes    No   No   No   Yes     Fast   Passive
      ge-1/0/4       Actor    No   Yes    No   No   No   Yes     Fast    Active
      ge-1/0/4     Partner    No   Yes    No   No   No   Yes     Fast   Passive
    LACP protocol:        Receive State  Transmit State          Mux State
      ge-1/0/1            Port disabled     No periodic           Detached
      ge-1/0/2                Defaulted   Fast periodic           Detached
      ge-1/0/3                Defaulted   Fast periodic           Detached
      ge-1/0/4                Defaulted   Fast periodic           Detached

{master}[edit]
ibm@J08E-re0# run show lacp statistics interfaces
Aggregated interface: ae0
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      ge-1/0/1                   0           1            0            0
      ge-1/0/2                   0          84            0            0
      ge-1/0/3                   0          84            0            0
      ge-1/0/4                   0          84            0            0

{master}[edit]
ibm@J08E-re0#
```

As indicated in the output of the **show** commands from Example 9-3 on page 233, the LAG is inactive because we enabled LACP only on one side of the LAG. The LACP is only active on one side of the LAG as you can see in the output of the **show lacp statistics interfaces** command. We used the same commands to configure LACP on the other side of the LAG on J48E. Example 9-4 shows how to verify, on the J08E switch, that the LAG is up and that LACP communication exists between the LAG peers.

Example 9-4 Verifying that LAG is up and LACP communication exists on both LAG sides

```
{master}[edit]
ibm@J08E-re0# run clear lacp statistics interfaces

{master}[edit]
ibm@J08E-re0# run show interfaces ae0 extensive
Physical interface: ae0, Enabled, Physical link is Up <---- LAG interface is up
  Interface index: 234, SNMP ifIndex: 508, Generation: 237
  Link-level type: Ethernet, MTU: 1514, Speed: 3Gbps, BPDU Error: None,
MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled, Flow
control: Disabled,
  Minimum links needed: 3, Minimum bandwidth needed: 0
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x0
  Current address: 00:26:88:87:e2:01, Hardware address: 00:26:88:87:e2:01
  Last flapped   : 2010-05-06 21:29:03 UTC (16:45:29 ago)
```

```
  Statistics last cleared: 2010-05-07 14:05:21 UTC (00:09:11 ago)
  Traffic statistics:
   Input  bytes  :               223890                 3072 bps
   Output bytes  :               242659                 3584 bps
   Input  packets:                 1704                    3 pps
   Output packets:                 1983                    4 pps
   IPv6 transit statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0, Policed discards:
0, Resource errors: 0
  Output errors:
    Carrier transitions: 0, Errors: 0, Drops: 0, MTU errors: 0, Resource errors: 0

  Logical interface ae0.0 (Index 65) (SNMP ifIndex 509) (Generation 138)
    Flags: SNMP-Traps 0x0 Encapsulation: ENET2
    Statistics        Packets        pps         Bytes          bps
    Bundle:
        Input :             0          0             0            0
        Output:           275          0         28325            0
    Link:
      ge-1/0/1.0 <-- down
        Input :             0          0             0            0
        Output:             0          0             0            0
      ge-1/0/2.0
        Input :             0          0             0            0
        Output:           294          0         32904            0
      ge-1/0/3.0
        Input :             0          0             0            0
        Output:            18          0          4338            0
      ge-1/0/4.0
        Input :             0          0             0            0
        Output:            18          0          4338            0
    LACP info:        Role     System             System      Port    Port  Port
                                priority          identifier  priority  number   key
      ge-1/0/1.0     Actor          0  00:26:88:87:e3:f0       127     577     1
      ge-1/0/1.0   Partner          0  00:00:00:00:00:00         0       0     0
      ge-1/0/2.0     Actor          0  00:26:88:87:e3:f0       127     578     1
      ge-1/0/2.0   Partner        127  00:19:e2:57:3a:00       127     642     2
      ge-1/0/3.0     Actor          0  00:26:88:87:e3:f0       127     579     1
      ge-1/0/3.0   Partner        127  00:19:e2:57:3a:00       127     643     2
      ge-1/0/4.0     Actor          0  00:26:88:87:e3:f0       127     580     1
      ge-1/0/4.0   Partner        127  00:19:e2:57:3a:00       127     644     2
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      ge-1/0/1.0                 0           0            0            0
      ge-1/0/2.0               551         550            0            0
      ge-1/0/3.0               551         550            0            0
      ge-1/0/4.0               551         550            0            0
    Marker Statistics:   Marker Rx     Resp Tx   Unknown Rx   Illegal Rx
      ge-1/0/1.0                 0           0            0            0
      ge-1/0/2.0                 0           0            0            0
      ge-1/0/3.0                 0           0            0            0
```

```
      ge-1/0/4.0                 0           0            0            0
    Protocol eth-switch, Generation: 157, Route table: 0
      Flags: None

{master}[edit]
ibm@J08E-re0# run show lacp interfaces
Aggregated interface: ae0
    LACP state:       Role   Exp   Def  Dist  Col  Syn  Aggr  Timeout  Activity
      ge-1/0/1       Actor    No   Yes    No   No   No   Yes     Fast    Active
      ge-1/0/1     Partner    No   Yes    No   No   No   Yes     Fast   Passive
      ge-1/0/2       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-1/0/2     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-1/0/3       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-1/0/3     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-1/0/4       Actor    No    No   Yes  Yes  Yes   Yes     Fast    Active
      ge-1/0/4     Partner    No    No   Yes  Yes  Yes   Yes     Fast    Active
    LACP protocol:        Receive State  Transmit State          Mux State
      ge-1/0/1            Port disabled     No periodic           Detached
      ge-1/0/2                  Current   Fast periodic Collecting distributing
      ge-1/0/3                  Current   Fast periodic Collecting distributing
      ge-1/0/4                  Current   Fast periodic Collecting distributing

{master}[edit]
ibm@J08E-re0# run show lacp statistics interfaces
Aggregated interface: ae0
    LACP Statistics:       LACP Rx     LACP Tx   Unknown Rx   Illegal Rx
      ge-1/0/1                  0           0            0            0
      ge-1/0/2                 49          49            0            0
      ge-1/0/3                 49          49            0            0
      ge-1/0/4                 49          49            0            0

{master}[edit]
ibm@J08E-re0#
```

As shown in Example 9-4 on page 235, the `ae0` LAG is up, with both Tx and Rx LACP packets. Therefore, the LACP is working on both sides of the LAG, and all member interfaces are active except the one that we disabled in Example 9-2 on page 230. We left this interface disabled on purpose to observe the difference between an active interface and a disabled or non-connected interface.

9.3 Configuring a VLAN

You can create and manage virtual local area networks (VLANs) on IBM j-type Ethernet switches as explained in this section. IBM j-type Ethernet routers are Layer 3 devices. We do not explain VLAN management on these devices.

IBM j-series switches have an untagged factory default VLAN named *default*. On an IBM J48E switch with a factory default configuration, switching is enabled on all interfaces. In addition, a VLAN named *default* is created, and all interfaces are placed into this VLAN.

To create a VLAN on an IBM j-type switch, use the following command:

```
{master}[edit]
ibm@J08E-re0# set vlans vlan_name vlan-id vlan_id_between-1..4094
```

where:

vlan-name is the name assigned to the VLAN.
vlan-id is the 802.1Q VLAN tag.

VLANs 0 and 4095 are reserved by Junos. Therefore, you cannot use them in your network.

To view the VLAN configuration, use the following command:

```
{master}[edit]
ibm@J08E-re0# show vlans
```

To view the configuration of a specific VLAN, use the following command:

```
{master}[edit]
ibm@J08E-re0# show vlans vlan_name
```

To view the status of the VLAN after you committed the configuration, use the following command:

```
{master}
ibm@J08E-re0> show vlans [vlan_name] [ brief | extensive ]
```

Example 9-5 shows the configuration and verification of VLANs, users, servers, and voice on a J08E switch. We use 100, 200, and 300 as 802.1Q VLAN tags for these VLANs.

Example 9-5 VLAN configuration and verification

```
{master}[edit]
ibm@J08E-re0# set vlans Users vlan-id 100

{master}[edit]
ibm@J08E-re0# set vlans Servers vlan-id 200

{master}[edit]
ibm@J08E-re0# set vlans Voice vlan-id 300

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# show vlans
Servers {
    vlan-id 200;
}
Users {
    vlan-id 100;
}
Voice {
    vlan-id 300;
}

{master}[edit]
```

```
ibm@J08E-re0# run show vlans
Name           Tag     Interfaces
Servers        200
                       None
Users          100
                       None
Voice          300
                       None
default
                       None

{master}[edit]
ibm@J08E-re0# run show vlans brief
                                                        Ports
Name           Tag       Primary Address             Active/Total
Servers        200
Users          100
Voice          300
default

{master}[edit]
ibm@J08E-re0# run show vlans Users extensive
VLAN: Users, Created at: Fri May  7 18:42:54 2010
802.1Q Tag: 100, Internal index: 6, Admin State: Enabled, Origin: Static
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 0 (Active = 0), Untagged  0 (Active = 0)

{master}[edit]
ibm@J08E-re0#
```

9.4 Configuring a port VLAN

Switch ports operate in either access mode or trunk mode. This section explains how to configure access and trunk ports on IBM j-type Ethernet switches.

9.4.1 Access ports

All ports default to access mode in the factory default configuration. You assign traffic to a particular VLAN by using either of the following methods:

- An interface (port) on the switch. You specify that all traffic received on a particular interface on the switch is assigned to a specific VLAN. If you use the default factory switch settings, all traffic received on an access interface is untagged. This traffic is part of a default VLAN, but it is not tagged with an 802.1Q tag. When configuring the switch, you specify which VLAN to assign the traffic to. You configure the VLAN either by using a VLAN number (called a VLAN ID) or by using a name, which the switch translates into a numeric VLAN ID.
- A Media Access Control (MAC) address. You can specify to forward all traffic received from a specific MAC address to a specific egress interface (next hop) on the switch. This method can be useful when you are using automated databases to manage the switches on your network. However, it is administratively cumbersome to configure manually and is not presented here.

To change the access VLAN of a port, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces type-fpc/pic/port.unit family ethernet-switching vlan
members vlan-name
```

where:

type-fpc/pic/port.unit	is the logical interface we want to configure
vlan-name	is the name of the VLAN or VLAN ID.

To see interface status and VLAN assignment, use the following command:

```
{master}
ibm@J08E-re0> show ethernet-switching interfaces [ type-fpc/pic/port.unit ]
```

To see the switching table (MAC table), use the following command:

```
{master}
ibm@J08E-re0> show ethernet-switching table [ brief | extensive | summary ]
```

To clear the MAC table, use the following command:

```
{master}
ibm@J08E-re0> clear ethernet-switching table
```

Example 9-6 shows how to create VLANs `Users`, `Servers`, and `Voice` with a VLAN ID of 100, 200, and 300. This example then assign ports ge-1/0/11.0 and ge-1/0/12.0 to the Users VLAN and ports ge-1/0/13.0 and ge-1/0/14.0 to the Servers VLAN. It also shows how to move in the Junos configuration hierarchy to shorten the command length.

Example 9-6 Assigning ports to VLANs

```
{master}[edit]
ibm@J08E-re0# edit vlans <------------ move in Junos configuration hierarchy
{master}[edit vlans]
ibm@J08E-re0# set Users vlan-id 100
{master}[edit vlans]
ibm@J08E-re0# set Servers vlan-id 200
{master}[edit vlans]
ibm@J08E-re0# set Voice vlan-id 300
{master}[edit vlans]
ibm@J08E-re0# show
Servers {
    vlan-id 200;
}
Users {
    vlan-id 100;
}
Voice {
    vlan-id 300;
}

{master}[edit vlans]
ibm@J08E-re0# exit
{master}[edit] <------- We returned to the previous configuration hierarchy level
ibm@J08E-re0#
{master}[edit]
ibm@J08E-re0# edit interfaces <-------- Move in the Junos configuration hierarchy
{master}[edit interfaces]
```

```
ibm@J08E-re0# set ge-1/0/11.0 family ethernet-switching vlan members Users
{master}[edit interfaces]
ibm@J08E-re0# set ge-1/0/12.0 family ethernet-switching vlan members 100
{master}[edit interfaces]
ibm@J08E-re0# set ge-1/0/13.0 family ethernet-switching vlan members Servers
{master}[edit interfaces]
ibm@J08E-re0# set ge-1/0/14.0 family ethernet-switching vlan members 200
{master}[edit interfaces]
ibm@J08E-re0# show
ge-1/0/11 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members Users;
            }
        }
    }
}
ge-1/0/12 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members 100;
            }
        }
    }
}
ge-1/0/13 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members Servers;
            }
        }
    }
}
ge-1/0/14 {
    unit 0 {
        family ethernet-switching {
            vlan {
                members 200;
            }
        }
    }
}

{master}[edit interfaces]
ibm@J08E-re0# top
{master}[edit] <---------- We returned to the top configuration hierarchy level
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
```

```
commit complete

{master}[edit]
ibm@J08E-re0# run show vlans brief
                                                        Ports
Name           Tag       Primary Address             Active/Total
Servers        200                                      0/2
Users          100                                      0/2
Voice          300
default

{master}[edit]
ibm@J08E-re0# run show vlans
Name           Tag     Interfaces
Servers        200
                       ge-1/0/13.0, ge-1/0/14.0
Users          100
                       ge-1/0/11.0, ge-1/0/12.0
Voice          300
                       None
default
                       None
{master}[edit]
ibm@J08E-re0#
```

9.4.2 Trunk ports

Interfaces configured as a trunk are typically uplinks to other switches or edge routers. To configure an interface as a trunk on an IBM j-type switch, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces type-fpc/pic/port.unit family ethernet-switching
port-mode trunk
```

where *type-fpc/pic/port.unit* is the logical interface to configure and as a trunk.

To assign and allow VLANs on a trunk interface on an IBM j-type switch, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces type-fpc/pic/port.unit family ethernet-switching vlan
members vlan-names
```

where:

type-fpc/pic/port.unit	is the trunk interface.
vlan-names	are the VLANs assigned on the trunk.

To specify more than one VLAN, enclose them in square brackets, for example `[ Vlan1 vlan2 vlanX ]`. To specify all VLANs, use the keyword `all`.

To specify which VLAN is existing or untagged on a trunk, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces type-fpc/pic/port.unit family ethernet-switching
native-vlan-id vlan-name
```

where:

type-fpc/pic/port.unit	is the trunk interface.
vlan-name	is the existing or untagged VLAN.

Example 9-7 shows the configuration of trunk port xe-3/0/0 and xe-3/0/1. On xe-3/0/0, all VLANs are allowed, and the existing or untagged VLAN is the default VLAN. On xe-3/0/1, only the Users and Servers VLANs are allowed. Again, this example shows moving deeper into the Junos configuration hierarchy to shorten the command length.

Example 9-7 Configuring and verifying trunk ports

```
{master}[edit]
ibm@J08E-re0# edit interfaces xe-3/0/0.0 family ethernet-switching

{master}[edit interfaces xe-3/0/0 unit 0 family ethernet-switching]
ibm@J08E-re0# set port-mode trunk

{master}[edit interfaces xe-3/0/0 unit 0 family ethernet-switching]
ibm@J08E-re0# set vlan members all

{master}[edit interfaces xe-3/0/0 unit 0 family ethernet-switching]
ibm@J08E-re0# set native-vlan-id default

{master}[edit interfaces xe-3/0/0 unit 0 family ethernet-switching]
ibm@J08E-re0# show
port-mode trunk;
vlan {
    members all;
}
native-vlan-id default;

{master}[edit interfaces xe-3/0/0 unit 0 family ethernet-switching]
ibm@J08E-re0# top edit interfaces xe-3/0/1.0 family ethernet-switching

{master}[edit interfaces xe-3/0/1 unit 0 family ethernet-switching]
ibm@J08E-re0# set port-mode trunk

{master}[edit interfaces xe-3/0/1 unit 0 family ethernet-switching]
ibm@J08E-re0# set vlan members [ Users Servers ]

{master}[edit interfaces xe-3/0/1 unit 0 family ethernet-switching]
ibm@J08E-re0# show
port-mode trunk;
vlan {
    members [ Users Servers ];
}

{master}[edit interfaces xe-3/0/1 unit 0 family ethernet-switching]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete
```

```
{master}[edit interfaces xe-3/0/1 unit 0 family ethernet-switching]
ibm@J08E-re0# exit

{master}[edit]
ibm@J08E-re0# run show vlans
Name           Tag     Interfaces
Servers        200
                       ge-1/0/13.0, ge-1/0/14.0, xe-3/0/0.0*, xe-3/0/1.0*
Users          100
                       ge-1/0/11.0, ge-1/0/12.0, xe-3/0/0.0*, xe-3/0/1.0*
Voice          300
                       xe-3/0/0.0*
default
                       xe-3/0/0.0*

{master}[edit]
ibm@J08E-re0# run show vlans brief
                                                          Ports
Name           Tag       Primary Address             Active/Total
Servers        200                                       2/4
Users          100                                       2/4
Voice          300                                       1/1
default                                                  1/1

{master}[edit]
ibm@J08E-re0# run show vlans extensive
VLAN: Servers, Created at: Fri May  7 18:42:54 2010
802.1Q Tag: 200, Internal index: 5, Admin State: Enabled, Origin: Static
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged  2 (Active = 0)
      xe-3/0/0.0*, tagged, trunk
      xe-3/0/1.0*, tagged, trunk
      ge-1/0/13.0, untagged, access
      ge-1/0/14.0, untagged, access

VLAN: Users, Created at: Fri May  7 18:42:54 2010
802.1Q Tag: 100, Internal index: 6, Admin State: Enabled, Origin: Static
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 2 (Active = 2), Untagged  2 (Active = 0)
      xe-3/0/0.0*, tagged, trunk
      xe-3/0/1.0*, tagged, trunk
      ge-1/0/11.0, untagged, access
      ge-1/0/12.0, untagged, access

VLAN: Voice, Created at: Fri May  7 18:42:54 2010
802.1Q Tag: 300, Internal index: 7, Admin State: Enabled, Origin: Static
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 1 (Active = 1), Untagged  0 (Active = 0)
      xe-3/0/0.0*, tagged, trunk

VLAN: default, Created at: Tue Apr 27 15:02:35 2010
Internal index: 2, Admin State: Enabled, Origin: Static
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 0 (Active = 0), Untagged  1 (Active = 1)
      xe-3/0/0.0*, untagged, trunk
```

```
{master}[edit]
ibm@J08E-re0# run show ethernet-switching interfaces
Interface    State  VLAN members        Tag   Tagging  Blocking
ge-1/0/11.0  down   Users               100   untagged blocked by STP
ge-1/0/12.0  down   Users               100   untagged blocked by STP
ge-1/0/13.0  down   Servers             200   untagged blocked by STP
ge-1/0/14.0  down   Servers             200   untagged blocked by STP
xe-3/0/0.0   up     default                   untagged unblocked
                    Servers             200   tagged   unblocked
                    Users               100   tagged   unblocked
                    Voice               300   tagged   unblocked
xe-3/0/1.0   up     Servers             200   tagged   blocked by STP
                    Users               100   tagged   blocked by STP
{master}[edit]
ibm@J08E-re0# exit
Exiting configuration mode

{master}
ibm@J08E-re0> show ethernet-switching interfaces
Interface    State  VLAN members        Tag   Tagging  Blocking
ge-1/0/11.0  down   Users               100   untagged blocked by STP
ge-1/0/12.0  down   Users               100   untagged blocked by STP
ge-1/0/13.0  down   Servers             200   untagged blocked by STP
ge-1/0/14.0  down   Servers             200   untagged blocked by STP
xe-3/0/0.0   up     default                   untagged unblocked
                    Servers             200   tagged   unblocked
                    Users               100   tagged   unblocked
                    Voice               300   tagged   unblocked
xe-3/0/1.0   up     Servers             200   tagged   blocked by STP
                    Users               100   tagged   blocked by STP

{master}
ibm@J08E-re0>
```

9.4.3 Routed VLAN interface

In a traditional network, broadcast domains consist of physical interfaces connected to a single switch or logical interfaces connected to one or more switches through VLAN configurations. Switches send traffic to hosts that are part of the same broadcast domain. However, routers are required to route traffic from one broadcast domain to another and to perform other Layer 3 functions such as traffic engineering. IBM j-type Ethernet switches use a Layer 3 routed VLAN interface (RVI), named *vlan*, to perform these routing functions to route data to other Layer 3 interfaces. The RVI functions as a logical router, which eliminates the need for having both a switch and a router. For redundancy, an RVI can be combined with implementations of the Virtual Router Redundancy Protocol (VRRP) in both bridging and Virtual Private LAN Service (VPLS) environments.

The RVI must be configured as part of a broadcast domain or VPLS routing instance for Layer 3 traffic to be routed out of it. The RVI supports IPv4, IPv6, Multiprotocol Label Switching (MPLS), and IS-IS traffic. At least one Layer 2 logical interface must be operational for the RVI to be operational. You must configure a broadcast domain or VPLS routing instance for the RVI just as you might configure a VLAN on the switch. Multicast data, broadcast data, or unicast data is switched between ports within the same RVI broadcast

domain or VPLS routing instance. The RVI routes data that is destined for the MAC address of the switch.

Jumbo frames of up to 9,216 bytes are supported on an RVI. To route jumbo data packets on the RVI, configure the jumbo MTU size on the member physical interfaces of the RVI and not on the RVI itself (the vlan interface). However, for jumbo control packets, for example, to ping the RVI with a packet size of 6000 bytes or more, you must explicitly configure the jumbo MTU size on the vlan interface (the RVI).

To configure an RVI, perform the following steps:

1. Create a Layer 2 VLAN by assigning it a name and a VLAN ID with the following command:

   ```
   {master}[edit]
   ibm@J08E-re0# set vlans vlan-name vlan-id vlan-id
   ```

2. Assign an interface to the VLAN by using the following command:

   ```
   {master}[edit]
   ibm@J08E-re0# set interfaces type-fpc/pic/port.unit family ethernet-switching
   vlan members vlan-name
   ```

3. Create a logical Layer 3 RVI on a subnet for the broadcast domain of the VLAN by using the following command:

   ```
   {master}[edit]
   ibm@J08E-re0# set interfaces vlan.unit family inet address IP/mask
   ```

4. Link the Layer 2 VLAN to the logical Layer 3 interface by using the following command:

   ```
   [edit]
   user@switch# set vlans vlan-name l3-interface vlan.unit
   ```

Example 9-8 shows how to configure an RVI using a new VLAN named `RV`, with `vlan-id 500`, and binding with Layer 3 interface `vlan.555`. A different number for `vlan-id` and the unit of the vlan interface are used to clearly illustrate the linking process.

Example 9-8 RVI configuration

```
{master}[edit]
ibm@J08E-re0# set vlans RV vlan-id 500

{master}[edit]
ibm@J08E-re0# set interfaces ge-1/0/15.0 family ethernet-switching vlan members RV

{master}[edit]
ibm@J08E-re0# set interfaces vlan.555 family inet address 10.1.1.1/24

{master}[edit]
ibm@J08E-re0# set vlans RV l3-interface vlan.555

{master}[edit]
ibm@J08E-re0#commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete
```

```
{master}[edit]
ibm@J08E-re0# exit
Exiting configuration mode

{master}
ibm@J08E-re0> show interfaces vlan terse
Interface               Admin Link Proto    Local                 Remote
vlan                    up    up
vlan.555                up    up   inet     10.1.1.1/24

{master}
ibm@J08E-re0> show vlans
Name           Tag     Interfaces
RV             500
                       ge-1/0/15.0
default
{master}
ibm@J08E-re0> show ethernet-switching table
Ethernet-switching table: 3 entries, 0 learned
  VLAN              MAC address       Type         Age Interfaces
  default           *                 Flood          - All-members
  RV                *                 Flood          - All-members
  RV                00:26:88:87:e2:00 Static         - Router

{master}
ibm@J08E-re0> show vlans RV extensive
VLAN: RV, Created at: Sun May  9 02:43:15 2010
802.1Q Tag: 500, Internal index: 9, Admin State: Enabled, Origin: Static
Layer 3 interface: vlan.555 (UP) <----------------- link to L3 interface
 IPV4 addresses:
                10.1.1.1/24(Primary)
Protocol: Port Mode, Mac aging time: 300 seconds
Number of interfaces: Tagged 0 (Active = 0), Untagged  1 (Active = 0)
      ge-1/0/15.0, untagged, access

{master}
ibm@J08E-re0>
```

9.5 Configuring the Spanning Tree Protocol

IBM j-type Ethernet switches provide Layer 2 loop prevention through Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), Multiple Spanning Tree Protocol (MSTP), and VLAN Spanning Tree Protocol (VSTP). The default STP is RSTP.

You use the following same operational commands to check the status of your spanning-tree configuration, regardless of which STP is configured:

```
{master}
ibm@J08E-re0> show spanning-tree bridge

{master}
ibm@J08E-re0> show spanning-tree interface
```

```
{master}
ibm@J08E-re0> show spanning-tree statistics interface
```

9.5.1 Spanning Tree Protocol

If your network includes 802.1D 1998 bridges, you can remove RSTP and explicitly configure STP. When you explicitly configure STP, the IBM j-type Ethernet switches use the IEEE 802.1D 2004 specification, force version 0. This configuration runs a version of RSTP that is compatible with the classic, basic STP. To configure STP, use the following commands:

```
{master}[edit]
ibm@J08E-re0# delete protocols rstp

{master}[edit]
ibm@J08E-re0# set protocols stp
```

You can further customize STP timers and options under the STP protocol configuration hierarchy as shown in Example 9-9.

Example 9-9 STP configuration

```
{master}[edit]
ibm@J08E-re0# delete protocols rstp

{master}[edit]
ibm@J08E-re0# set protocols stp

{master}[edit]
ibm@J08E-re0# edit protocols stp

{master}[edit protocols stp]
ibm@J08E-re0# set hello-time 2

{master}[edit protocols stp]
ibm@J08E-re0# set max-age 20

{master}[edit protocols stp]
ibm@J08E-re0# set forward-delay 15

{master}[edit protocols stp]
ibm@J08E-re0# set bridge-priority 32k

{master}[edit protocols stp]
ibm@J08E-re0# set interface xe-3/0/0.0 priority 64

{master}[edit protocols stp]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit protocols stp]
ibm@J08E-re0# run show spanning-tree bridge
```

```
STP bridge parameters
Context ID                          : 0
Enabled protocol                    : STP <---------- Note the STP
  Root ID                           : 32768.00:26:88:87:e2:00
  Hello time                        : 2 seconds
  Maximum age                       : 20 seconds
  Forward delay                     : 15 seconds
  Message age                       : 0
  Number of topology changes        : 1
  Time since last topology change   : 950 seconds
  Topology change initiator         : xe-3/0/0.0
  Topology change last recvd. from  : 00:26:88:87:e2:91
  Local parameters
    Bridge ID                       : 32768.00:26:88:87:e2:00
    Extended system ID              : 0
    Internal instance ID            : 0

{master}[edit protocols stp]
ibm@J08E-re0# run show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
xe-3/0/0.0      64:849       64:849  32768.00268887e200      2000  FWD    DESG
xe-3/0/1.0     128:850       64:849  32768.00268887e200      2000  BLK    BKUP

{master}[edit protocols stp]
ibm@J08E-re0#
```

9.5.2 Rapid Spanning Tree Protocol

By default, IBM j-type Ethernet switches use the RSTP to provide better reconvergence time than the original STP. Switches configured for STP and RSTP interoperate with one another. However, you must keep in mind a few basic considerations.

For example, if a switch supports only STP and interconnects with a switch running RSTP, that switch discards the RSTP Bridge Protocol Data Units (BPDU). The RSTP-capable switch, upon receiving STP BPDUs, reverts to STP mode, thus allowing interoperability between the two devices.

If the default configuration was removed, you can configure RSTP by using the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rstp
```

You can further customize RSTP timers and options under the RSTP configuration hierarchy as shown in Example 9-10.

Example 9-10 RSTP configuration

```
{master}[edit]
ibm@J08E-re0# set protocols rstp

{master}[edit]
ibm@J08E-re0# edit protocols rstp

{master}[edit protocols rstp]
ibm@J08E-re0# set hello-time 2

{master}[edit protocols rstp]
ibm@J08E-re0# set max-age 20

{master}[edit protocols rstp]
ibm@J08E-re0# set forward-delay 15

{master}[edit protocols rstp]
ibm@J08E-re0# set bridge-priority 32k

{master}[edit protocols rstp]
ibm@J08E-re0# set interface xe-3/0/0.0 priority 64

{master}[edit protocols rstp]
ibm@J08E-re0# set interface xe-3/0/1.0 mode point-to-point

{master}[edit protocols rstp]
ibm@J08E-re0# show
bridge-priority 32k;
max-age 20;
hello-time 2;
forward-delay 15;
interface xe-3/0/0.0 {
    priority 64;
}
interface xe-3/0/1.0 {
    mode point-to-point;
}

{master}[edit protocols rstp]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit protocols rstp]
ibm@J08E-re0# run show spanning-tree bridge
STP bridge parameters
Context ID                          : 0
Enabled protocol                    : RSTP <---------- note the RSTP
```

```
  Root ID                           : 32768.00:26:88:87:e2:00
  Hello time                        : 2 seconds
  Maximum age                       : 20 seconds
  Forward delay                     : 15 seconds
  Message age                       : 0
  Number of topology changes        : 1
  Time since last topology change   : 16 seconds
  Topology change initiator         : xe-3/0/0.0
  Topology change last recvd. from  : 00:26:88:87:e2:91
  Local parameters
    Bridge ID                       : 32768.00:26:88:87:e2:00
    Extended system ID              : 0
    Internal instance ID            : 0

{master}[edit protocols rstp]
ibm@J08E-re0# run show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
xe-3/0/0.0      64:849       64:849  32768.00268887e200      2000  FWD    DESG
xe-3/0/1.0     128:850       64:849  32768.00268887e200      2000  BLK    BKUP

{master}[edit protocols rstp]
ibm@J08E-re0# run show spanning-tree interface detail

Interface name                 : xe-3/0/0.0
Port identifier                : 64.849
Designated port ID             : 64.849
Port cost                      : 2000
Port state                     : Forwarding
Designated bridge ID           : 32768.00:26:88:87:e2:00
Port role                      : Designated
Link type                      : Pt-Pt/NONEDGE
Boundary port                   : NA

Interface name                 : xe-3/0/1.0
Port identifier                : 128.850
Designated port ID             : 64.849
Port cost                      : 2000
Port state                     : Blocking
Designated bridge ID           : 32768.00:26:88:87:e2:00
Port role                      : Backup
Link type                      : Pt-Pt/NONEDGE
Boundary port                   : NA


{master}[edit protocols rstp]
ibm@J08E-re0# run show spanning-tree statistics interface


Interface   BPDUs sent   BPDUs received      Next BPDU
                                             transmission
xe-3/0/0.0        2647             1                0
```

```
xe-3/0/1.0          1          2647                0

{master}[[edit protocols rstp]
ibm@J08E-re0#
```

9.5.3 Multiple Spanning Tree Protocol

Although RSTP provides a faster convergence time than STP, it does not solve the inherent problem in STP: All VLANs within a LAN must share the same spanning tree. To solve this problem, we use the MSTP to create a loop-free topology in networks with multiple spanning-tree regions.

With an MSTP region, a group of bridges can be modeled as a single bridge. An MSTP region contains multiple spanning tree instances (MSTIs). MSTIs provide different paths for different VLANs. This functionality facilitates better load sharing across redundant links. An MSTP region can support up to 64 MSTIs. Each instance can support anywhere from 1 through 4094 VLANs.

Because the MSTP encodes region information after the standard RSTP BPDU, a switch running the RSTP interprets the MSTP BPDUs as RSTP BPDUs. This behavior facilitates full compatibility between devices running MSTP and devices running the STP or RSTP. All RSTP switches outside of an MST region view the MST region as a single RSTP switch.

The *common spanning tree* (CST) interconnects all MST regions and STP devices that are not bound to a particular region. It facilitates end-to-end paths within an MSTP environment. All MSTP environments contain a CST, which is used to interconnect individual MST regions and independent STP devices. A single root bridge is elected and tasked with path calculation for the CST. Each MST region is treated as a virtual bridge within the environment, regardless of the number of devices that participate in the MST region.

The *common and internal spanning tree* (CIST) is a single topology that connects all switches (STP, RSTP, and MSTP devices) through an active topology. The CIST includes a single spanning tree as calculated by the STP and RSTP with the logical continuation of connectivity through MST regions. The CIST is calculated by MSTP and ensures connectivity between LANs and devices within a bridged network.

To create an MSTP region and assign a revision level use the following commands:

```
{master}[edit]
ibm@J08E-re0# set protocols mstp configuration-name region_name

{master}[edit]
ibm@J08E-re0# set protocols mstp revision-level revision_level
```

where:

region_name is the name of the MSTP region.
revision_level is the revision_level of the MSTP region.

The *region_name* and *revision_level* must mach on all routers in the same region.

To create an MSTI in an MSTP region and assign VLANs to it, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols mstp msti msti_id vlan vlans
```

where:

msti_id	is the ID of the MSTI.
vlans	refers to a list of VLANS that are part of this specific MSTI.

To inspect the configuration of MSTP, use the following command:

```
{master}[edit]
ibm@J08E-re0> show spanning-tree mstp configuration
```

Example 9-11 shows a basic MSTP configuration and VLAN assignment.

Example 9-11 MSTP configuration

```
{master}[edit]
ibm@J08E-re0# set protocols mstp configuration-name region1

{master}[edit]
ibm@J08E-re0# set protocols mstp revision-level 1

{master}[edit]
ibm@J08E-re0# set protocols mstp msti 1 vlan Voice

{master}[edit]
ibm@J08E-re0# set protocols mstp msti 2 vlan [ Users Servers ]

{master}[edit]
ibm@J08E-re0# set protocols mstp msti 2 bridge-priority 32k

{master}[edit]
ibm@J08E-re0# set protocols mstp msti 2 interface xe-3/0/1.0 priority 64

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# show protocols mstp
configuration-name region1;
revision-level 1;
msti 1 {
    vlan Voice;
}
msti 2 {
    bridge-priority 32k;
    vlan [ Users Servers ];
    interface xe-3/0/1.0 {
        priority 64;
    }
}

{master}[edit]
```

```
ibm@J08E-re0#
ibm@J08E-re0# exit
Exiting configuration mode

{master}
ibm@J08E-re0> show spanning-tree mstp configuration
MSTP information
Context identifier     : 0
Region name            : region1
Revision               : 1
Configuration digest   : 0x69cc9b4dbdadc69662236e912f2f5cf0


MSTI     Member VLANs
   0 0-99,101-199,201-299,301-4094
   1 300
   2 100,200

{master}
ibm@J08E-re0> show spanning-tree bridge

STP bridge parameters
Context ID                          : 0
Enabled protocol                    : MSTP <--------- Note the MSTP

STP bridge parameters for CIST
  Root ID                           : 32768.00:26:88:87:e2:00
  CIST regional root                : 32768.00:26:88:87:e2:00
  CIST internal root cost           : 0
  Hello time                        : 2 seconds
  Maximum age                       : 20 seconds
  Forward delay                     : 15 seconds
  Number of topology changes        : 1
  Time since last topology change   : 385 seconds
  Topology change initiator         : xe-3/0/0.0
  Topology change last recvd. from  : 00:26:88:87:e2:91
  Local parameters
    Bridge ID                       : 32768.00:26:88:87:e2:00
    Extended system ID              : 0
    Internal instance ID            : 0

STP bridge parameters for MSTI 1
  MSTI regional root                : 32769.00:26:88:87:e2:00
  Hello time                        : 2 seconds
  Maximum age                       : 20 seconds
  Forward delay                     : 15 seconds
  Number of topology changes        : 1
  Topology change initiator         : xe-3/0/0.0
  Local parameters
    Bridge ID                       : 32769.00:26:88:87:e2:00
    Extended system ID              : 0
    Internal instance ID            : 1

STP bridge parameters for MSTI 2
  MSTI regional root                : 32770.00:26:88:87:e2:00
```

```
  Hello time                        : 2 seconds
  Maximum age                       : 20 seconds
  Forward delay                     : 15 seconds
  Number of topology changes        : 2
  Topology change initiator         : xe-3/0/1.0
  Topology change last recvd. from  : 00:26:88:87:e2:92
  Local parameters
    Bridge ID                       : 32770.00:26:88:87:e2:00
    Extended system ID              : 0
    Internal instance ID            : 2

{master}
ibm@J08E-re0> show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
xe-3/0/0.0     128:849      128:849  32768.00268887e200      2000  FWD    DESG
xe-3/0/1.0     128:850      128:849  32768.00268887e200      2000  BLK    BKUP

Spanning tree interface parameters for instance 1

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
xe-3/0/0.0     128:849      128:849  32769.00268887e200      2000  FWD    DESG

Spanning tree interface parameters for instance 2

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
xe-3/0/0.0     128:849       64:850  32770.00268887e200      2000  BLK    BKUP
xe-3/0/1.0      64:850       64:850  32770.00268887e200      2000  FWD    DESG

{master}
ibm@J08E-re0> show spanning-tree interface xe-3/0/0.0 detail

Spanning tree interface parameters for instance 0

Interface name                 : xe-3/0/0.0
Port identifier                : 128.849
Designated port ID             : 128.849
Port cost                      : 2000
Port state                     : Forwarding
Designated bridge ID           : 32768.00:26:88:87:e2:00
Port role                      : Designated
Link type                      : Pt-Pt/NONEDGE
Boundary port                   : No


Spanning tree interface parameters for instance 1

Interface name                 : xe-3/0/0.0
Port identifier                : 128.849
Designated port ID             : 128.849
```

```
Port cost                      : 2000
Port state                     : Forwarding
Designated bridge ID           : 32769.00:26:88:87:e2:00
Port role                      : Designated
Link type                      : Pt-Pt/NONEDGE
Boundary port                   : No


Spanning tree interface parameters for instance 2

Interface name                 : xe-3/0/0.0
Port identifier                : 128.849
Designated port ID             : 64.850
Port cost                      : 2000
Port state                     : Blocking
Designated bridge ID           : 32770.00:26:88:87:e2:00
Port role                      : Backup
Link type                      : Pt-Pt/NONEDGE
Boundary port                   : No


{master}
ibm@J08E-re0>
```

9.5.4 VLAN Spanning Tree Protocol

With the VSTP, IBM j-type Ethernet switches can run one or more STP or RSTP instances for each VLAN on which the VSTP is enabled. For networks with multiple VLANs, more intelligent tree spanning is possible, because each VLAN can have interfaces enabled or disabled depending on the available paths to that specific VLAN.

By default, the VSTP runs the RSTP, but the stand-alone RSTP and VSTP cannot run simultaneously on a switch. VSTP can be enabled for up to 253 VLANs.

To configure VSTP on all defined VLANs, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols vstp vlan all
```

To configure VSTP on a subset of defined VLANs, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols vstp vlan [vlan-list]
```

You can further customize RSTP timers and options under the VSTP configuration hierarchy as shown in Example 9-12.

Example 9-12 VSTP configuration

```
ibm@J08E-re0# set protocols vstp vlan all

{master}[edit]
ibm@J08E-re0# set protocols vstp vlan Users bridge-priority 32k

{master}[edit]
ibm@J08E-re0# edit protocols vstp vlan Servers
```

```
{master}[edit protocols vstp vlan Servers]
ibm@J08E-re0# set hello-time 2

{master}[edit protocols vstp vlan Servers]
ibm@J08E-re0# set max-age 20

{master}[edit protocols vstp vlan Servers]
ibm@J08E-re0# set forward-delay 15

{master}[edit protocols vstp vlan Servers]
ibm@J08E-re0# set interface xe-3/0/1.0 priority 64

{master}[edit protocols vstp vlan Servers]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit protocols vstp vlan Servers]
ibm@J08E-re0# top show protocols vstp
vlan Servers {
    max-age 20;
    hello-time 2;
    forward-delay 15;
    interface xe-3/0/1.0 {
        priority 64;
    }
}
vlan Users {
    bridge-priority 32k;
}
vlan all;

{master}[edit protocols vstp vlan Servers]
ibm@J08E-re0#
ibm@J08E-re0# run show spanning-tree bridge

STP bridge parameters
Context ID                          : 1
Enabled protocol                    : RSTP

STP bridge parameters for VLAN 200
  Root ID                           : 32968.00:26:88:87:e2:00
  Hello time                        : 2 seconds
  Maximum age                       : 20 seconds
  Forward delay                     : 15 seconds
  Message age                       : 0
  Number of topology changes        : 0
  Local parameters
    Bridge ID                       : 32968.00:26:88:87:e2:00
    Extended system ID              : 1
    Internal instance ID            : 0
```

```
STP bridge parameters
Context ID                          : 2
Enabled protocol                    : RSTP

STP bridge parameters for VLAN 100
  Root ID                           : 32868.00:26:88:87:e2:00
  Hello time                        : 2 seconds
  Maximum age                       : 20 seconds
  Forward delay                     : 15 seconds
  Message age                       : 0
  Number of topology changes        : 1
  Time since last topology change   : 459 seconds
  Topology change initiator         : xe-3/0/0.0
  Topology change last recvd. from  : 00:26:88:87:e2:91
  Local parameters
    Bridge ID                       : 32868.00:26:88:87:e2:00
    Extended system ID              : 2
    Internal instance ID            : 0

STP bridge parameters
Context ID                          : 3
Enabled protocol                    : RSTP

STP bridge parameters for VLAN 500
  Root ID                           : 33268.00:26:88:87:e2:00
  Hello time                        : 2 seconds
  Maximum age                       : 20 seconds
  Forward delay                     : 15 seconds
  Message age                       : 0
  Number of topology changes        : 1
  Time since last topology change   : 461 seconds
  Topology change initiator         : xe-3/0/0.0
  Topology change last recvd. from  : 00:26:88:87:e2:91
  Local parameters
    Bridge ID                       : 33268.00:26:88:87:e2:00
    Extended system ID              : 3
    Internal instance ID            : 0

STP bridge parameters
Context ID                          : 4
Enabled protocol                    : RSTP

STP bridge parameters for VLAN 300
  Root ID                           : 33068.00:26:88:87:e2:00
  Hello time                        : 2 seconds
  Maximum age                       : 20 seconds
  Forward delay                     : 15 seconds
  Message age                       : 0
  Number of topology changes        : 0
  Local parameters
    Bridge ID                       : 33068.00:26:88:87:e2:00
    Extended system ID              : 4
    Internal instance ID            : 0
```

```
{master}[edit protocols vstp vlan Servers]
ibm@J08E-re0# run show spanning-tree interface

Spanning tree interface parameters for VLAN 200

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
xe-3/0/0.0     128:849      128:849  32968.00268887e200      2000  FWD    DESG
xe-3/0/1.0      64:850       64:850  32968.00268887e200      2000  FWD    DESG

Spanning tree interface parameters for VLAN 100

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
xe-3/0/0.0     128:849      128:849  32868.00268887e200      2000  FWD    DESG
xe-3/0/1.0     128:850      128:849  32868.00268887e200      2000  BLK    BKUP

Spanning tree interface parameters for VLAN 500

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
xe-3/0/0.0     128:849      128:849  33268.00268887e200      2000  FWD    DESG

Spanning tree interface parameters for VLAN 300

Interface    Port ID    Designated      Designated         Port    State  Role
                         port ID        bridge ID          Cost
xe-3/0/0.0     128:849      128:849  33068.00268887e200      2000  FWD    DESG

{master}[edit protocols vstp vlan Servers]
ibm@J08E-re0#
```

9.5.5 BPDU protection

BPDU protection can help prevent STP misconfiguration that can lead to network outages. Receipt of BPDUs on certain interfaces in an STP, RSTP, VSTP, or MSTP topology can lead to network outages.

You enable BPDU protection on switch interfaces that are connected to user devices or on interfaces on which no BPDUs are expected, such as edge ports. If BPDUs are received on a protected interface, the interface is disabled and stops forwarding frames.

You can configure BPDU protection on a switch with or without a spanning tree. This type of topology typically consists of a non-STP switch connected to an STP switch through a trunk interface.

To configure BPDU protection on a switch with a spanning tree, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols <stp|rstp|mstp|vstp> bpdu-block-on-edge
```

To define the edge interface, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols <stp|rstp|mstp|vstp> interface interface edge
```

To configure BPDU protection on a switch without a spanning tree, use the following command:

```
{master}[edit]
ibm@J08E-re0# set ethernet-switching-options bpdu-block interfaces interface
```

After fixing the misconfiguration in the topology that triggered the BPDUs being sent to an interface, you can unblock the interface in either of the following ways:

- If the `disable-timeout` statement is included in the BPDU configuration, the interface automatically returns to service after the timer expires. To set this timer, use the following command:

  ```
  {master}[edit ethernet-switching-options]
  ibm@J08E-re0# set bpdu-block disable-timeout timeout interface interface
  ```

- Use the operational mode command:

  ```
  {master}
  ibm@J08E-re0> clear ethernet-switching bpdu-error
  ```

Disabling the BPDU protection configuration does not unblock the interface.

9.5.6 Loop protection

Loop protection increases the efficiency of the STP, RSTP, VSTP, and MSTP by preventing ports from moving into a forwarding state that might result in a loop opening in the network.

A blocking interface can make the transition to the forwarding state in error if the interface stops receiving BPDUs from its designated port on the segment. Such a transition error can occur when a hardware error happens on the switch or software configuration error between the switch and its neighbor.

When loop protection is enabled, the spanning-tree topology detects root ports and blocked ports and ensures that both keep receiving BPDUs. If a loop-protection-enabled interface stops receiving BPDUs from its designated port, it reacts as it might react to a problem with the physical connection on this interface. It does not change the interface to a forwarding state, but instead changes it to a loop-inconsistent state. The interface recovers, and then it changes back to the spanning-tree blocking state as soon as it receives a BPDU.

You must enable loop protection on all switch interfaces that have a chance of becoming root or designated ports. Loop protection is most effective when enabled in the entire switched network. When you enable loop protection, you must configure at least one action (**alarm**, **block**, or both).

An interface can be configured for either loop protection or root protection, but not for both.

To configure loop protection for an interface, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rstp interface interface bpdu-timeout-action block
```

To receive an alarm when loop protection fired on an interface, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rstp interface interface bpdu-timeout-action alarm
```

9.5.7 Root protection

Root protection increases the stability and security of the STP, RSTP, VSTP, and MSTP by limiting the ports that can be elected as root ports. A root port elected through the regular process has the possibility of being wrongly elected. A user bridge application running on a PC can also generate BPDUs and interfere with root port election. With root protection, network administrators can manually enforce the root bridge placement in the network.

Enable root protection on interfaces that must not receive superior BPDUs from the root bridge and that must not be elected as the root port. These interfaces become designated ports and are typically located on an administrative boundary. If the bridge receives superior STP BPDUs on a port that has root protection enabled, that port transitions to a root-prevented STP state (inconsistency state), and the interface is blocked. This blocking prevents a bridge that must not be the root bridge from being elected the root bridge. After the bridge stops receiving superior STP BPDUs on the interface with root protection, the interface returns to a listening state, followed by a learning state, and back to a forwarding state. Recovery back to the forwarding state is automatic.

When root protection is enabled on an interface, it is enabled for all the STP instances on that interface. The interface is blocked only for instances for which it receives superior BPDUs. Otherwise, it participates in the spanning-tree topology.

An interface can be configured for either root protection or loop protection, but not for both. To configure root protection for an interface, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rstp interface interface no-root-port
```

9.6 Link Layer Discovery Protocol

The Link Layer Discovery Protocol (LLDP) is used to learn and distribute device information on network links.

9.6.1 Understanding LLDP

IBM j-type Ethernet switches and routers use LLDP and LLDP–Media Endpoint Discovery (LLDP-MED) to learn and distribute device information on network links. With this information, the switch can quickly identify various devices, resulting in a LAN that interoperates smoothly and efficiently.

LLDP-capable devices transmit information in Type Length Value (TLV) messages to neighbor devices. Device information can include such specifics as chassis and port identification, system name, and system capabilities. The TLVs take this information from parameters that have already been configured in the Junos software.

LLDP-MED goes one step further by exchanging IP-telephony messages between the switch and the IP telephone. These TLV messages provide detailed information about the Power over Ethernet (PoE) policy. With the PoE Management TLVs, the switch ports can advertise the power level and power priority that are needed. For example, the switch can compare the power that is needed by an IP telephone running on a PoE interface with available resources. If the switch cannot meet the resources that are required by the IP telephone, the switch can negotiate with the telephone until a compromise on power is reached.

The switch also uses these protocols to ensure that voice traffic is tagged and prioritized with the correct values at the source itself. For example, 802.1p class-of-service (COS) and 802.1Q tag information can be sent to the IP telephone.

9.6.2 LLDP TLVs

IBM j-type Ethernet switches and routers support the following basic TLVs:

Chassis identifier The MAC address associated with the local system.

Port identifier The port identification for the specified port in the local system.

Port description The user-configured port description. The port description can be a maximum of 256 characters.

System name The user-configured name of the local system. The system name can be a maximum of 256 characters.

System description The system description containing information about the software and current image running on the system. This information is not configurable, but taken from the software.

System capabilities The primary function performed by the system. The capabilities that system supports, for example, a bridge or router. This information is not configurable, but is based on the model of the product.

Management address
The IP management address of the local system.

IBM j-type Ethernet switches and routers support the following 802.3 TLVs:

Power via MDI A TLV that advertises MDI power support, Power Sourcing Equipment (PSE) power pair, and power class information.

MAC/PHY configuration status
A TLV that advertises information about the physical interface, such as autonegotiation status and support and MAU type. The information is not configurable, but is based on the physical interface structure.

Link aggregation A TLV that advertises if the port is aggregated and its aggregated port ID.

Maximum frame size A TLV that advertises the Maximum Transmission Unit (MTU) of the interface that is sending LLDP frames.

Port VLAN A TLV that advertises the VLAN name configured on the interface.

IBM j-type Ethernet switches and routers support the following LLDP-MED TLVs:

LLDP MED capabilities
A TLV that advertises the primary function of the port. The capabilities values range 0 through 15:

0 Capabilities
1 Network Policy
2 Location Identification
3 Extended Power via MDI-PSE
4 Inventory
5–15 Reserved

LLDP-MED device class values

0 Class not defined.
1 Class 1 Device.

2 Class 2 Device.
3 Class 3 Device.
4 Network Connectivity Device
5–255 Reserved.

Network policy A TLV that advertises the port VLAN configuration and associated Layer 2 and Layer 3 attributes. Attributes include the policy identifier, application types, such as voice or streaming video, 802.1Q VLAN tagging, and 802.1p priority bits and Diffserv code points.

Endpoint location A TLV that advertises the physical location of the endpoint.

Extended power via MDI
A TLV that advertises the power type, power source, power priority, and power value of the port. The PSE device (network connectivity device) is responsible for advertising the power priority on a port.

9.6.3 Considerations for LLDP and LLDP-MED

Keep in mind the following considerations for using the LLDP and LLDP-MED:

- All mandatory and optional LLDP TLVs are advertised as soon as LLDP is enabled, but LLDP-MED TLVs are sent only after detecting an MED device on the link.
- Periodic LLDP and LLDP-MED updates are sent at a default interval of 30 seconds.
- Triggered updates are sent when the local value changes.
- Updates are sent as unsecure, one-way advertisements.
- If a switch port connects to an untrusted boundary, disable the LLDP and LLDP-MED on that port.
- When 802.1X authentication is enabled, LLDP frames are not transmitted or received until the port is authenticated. An IP phone and PC connected to the same switch port can be authenticated separately and can receive different VLAN assignments and policies for data and voice.

9.6.4 Configuring and monitoring LLDP

You can configure and monitor LLDP and LLDP-MED as explained in the following sections.

Configuring LLDP

LLDP is enabled on all interfaces by default. If it is disabled, you can enable LLDP by configuring it on all interfaces or specific interfaces. To configure LLDP on all interfaces or on a specific interface, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols lldp interface all
```

You can adjust the settings for LLDP advertisements for troubleshooting or verification purposes. For normal operations, do not adjust these settings from the default values.

To change the default 30 seconds interval at which LLDP advertisements are sent, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols lldp advertisement-interval 5..32768
```

You can change the default value of 4 for the multiplier that is used in combination with the advertisement-interval value to determine the length of time LLDP information is held before it is discarded. You can change this value by using the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols lldp hold-multiplier 2..10
```

Configuring LLDP-MED

LLDP-MED is enabled on all interfaces by default. If it is disabled, you can enable LLDP-MED by configuring it on all interfaces or on specific interfaces. To configure LLDP-MED on all interfaces or on a specific interface, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols lldp-med interface all
```

You can specify the number of LLDP-MED advertisements that are sent from the switch in the first second after it has detected an LLDP-capable device. The default is 3. To change the default value to another value, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols lldp-med fast-start 1..10
```

You can configure the location information that is advertised from the switch to the LLDP-MED device. You can specify a civic-based location (geographic location) or a location based on an emergency location identification string (ELIN). To specify a location by geography, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols lldp-med interface interface_name location
civic-based civic-options
```

For a detailed description of *civic-options,* see the appropriate documentation listed in the 9.7, “More information” on page 270. For a sample configuration of *civic-options*, see Example 9-13 on page 265.

To specify a location using an ELIN 10-digit number (area code and telephone number), use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols lldp-med interface interface_name location elin
10-digit-elin-number
```

Monitoring LLDP and LLDP-MED

This section shows the commands to verify and monitor the LLDP and LLDP-MED on IBM j-type Ethernet switches and routers.

To verify the LLDP and LLDP-MED status, use the following command:

```
{master}
ibm@J08E-re0> show lldp detail
```

To view learned neighbor information, use the following command:

```
{master}
ibm@J08E-re0> show lldp neighbors
```

To view detailed LLDP information about a neighbor, use the following command:

```
{master}
ibm@J08E-re0> show lldp neighbors interface interface-name
```

To check local LLDP and LLDP-MED details, use the following command:

```
{master}
ibm@J08E-re0> show lldp local-information
```

To view LLDP and LLDP-MED statistics and counters, use the following command:

```
{master}
ibm@J08E-re0> show lldp statistics
```

Example 9-13 shows the configuration and verification of LLDP and LLDP-MED on an IBM J08E switch.

Example 9-13 Configuration and verification of LLDP and LLDP-MED

```
{master}[edit]
ibm@J08E-re0# show protocols <------------ check if lldp and lldp-med are enabled
rstp;
lldp {
    interface all;
}
lldp-med {
    interface all;
}
{master}[edit]
ibm@J08E-re0# set protocols lldp advertisement-interval 5  <--speed up the updates

{master}[edit]
ibm@J08E-re0# set protocols lldp interface ge-1/0/1.0 disable

{master}[edit]
ibm@J08E-re0# edit protocols lldp-med

{master}[edit protocols lldp-med]
ibm@J08E-re0# set interface ge-1/0/2.0 location elin 4085551212

{master}[edit protocols lldp-med]
ibm@J08E-re0# edit interface ge-1/0/3.0 location civic-based

{master}[edit protocols lldp-med interface ge-1/0/3.0 location civic-based]
ibm@J08E-re0# set country-code US

{master}[edit protocols lldp-med interface ge-1/0/3.0 location civic-based]
ibm@J08E-re0# set ca-type 1 ca-value “El Dorado County”

{master}[edit protocols lldp-med interface ge-1/0/3.0 location civic-based]
ibm@J08E-re0# set ca-type 2 ca-value CA

{master}[edit protocols lldp-med interface ge-1/0/3.0 location civic-based]
ibm@J08E-re0# set ca-type 3 ca-value Somerset

{master}[edit protocols lldp-med interface ge-1/0/3.0 location civic-based]
ibm@J08E-re0# set ca-type 6 ca-value “Mount Aukum Road”

{master}[edit protocols lldp-med interface ge-1/0/3.0 location civic-based]
ibm@J08E-re0# set ca-type 19 ca-value 6450
```

```
{master}[edit protocols lldp-med interface ge-1/0/3.0 location civic-based]
ibm@J08E-re0# set ca-type 21 ca-value “Holiday Market”

{master}[edit protocols lldp-med interface ge-1/0/3.0 location civic-based]
ibm@J08E-re0# top

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# show protocols
rstp;
lldp {
    advertisement-interval 5;
    interface all;
    interface ge-1/0/1.0 {
        disable;
    }
}
lldp-med {
    interface all;
    interface ge-1/0/3.0 {
        location {
            civic-based {
                country-code US;
                ca-type {
                    1 {
                        ca-value "El Dorado County";
                    }
                    2 {
                        ca-value CA;
                    }
                    3 {
                        ca-value Somerset;
                    }
                    6 {
                        ca-value "Mount Aukum Road";
                    }
                    19 {
                        ca-value 6450;
                    }
                    21 {
                        ca-value "Holiday Market";
                    }
                }
            }
        }
    }
    interface ge-1/0/2.0 {
```

```
        location {
            elin 4085551212;
        }
    }
}

{master}[edit]
ibm@J08E-re0# exit
Exiting configuration mode

{master}
ibm@J08E-re0> show lldp detail

LLDP                   : Enabled
Advertisement interval : 5 seconds
Transmit delay         : 2 seconds
Hold timer             : 4 seconds
Notification interval  : 0 Second(s)
Config Trap Interval   : 0 seconds
Connection Hold timer  : 300 seconds

LLDP MED               : Enabled
MED fast start count   : 3 Packets


Interface      LLDP       LLDP-MED    Neighbor count
all            Enabled    Enabled     2

Interface     Vlan-id     Vlan-name
xe-3/0/0.0    0           default
xe-3/0/0.0    200         Servers
xe-3/0/0.0    100         Users
xe-3/0/0.0    300         Voice
xe-3/0/0.0    500         RV
xe-3/0/1.0    200         Servers
xe-3/0/1.0    100         Users
ge-1/0/15.0   500         RV
ge-1/0/11.0   100         Users
ge-1/0/12.0   100         Users
ge-1/0/13.0   200         Servers
ge-1/0/14.0   200         Servers

LLDP basic TLVs supported:
Chassis identifier, Port identifier, Port description, System name, System
description, System capabilities, Management address.

Supported LLDP 802 TLVs:
Power via MDI, Link aggregation, Maximum frame size, Port VLAN tag, Port
VLAN name.

Supported LLDP MED TLVs:
LLDP MED capabilities, Network policy, Endpoint location, Extended power
Via MDI.
```

```
{master}
ibm@J08E-re0> show lldp neighbors
LocalInterface Chassis Id        Port info     System Name
xe-3/0/0.0     00:26:88:87:e3:f0  xe-3/0/1.0   J08E-re0
xe-3/0/1.0     00:26:88:87:e3:f0  xe-3/0/0.0   J08E-re0

{master}
ibm@J08E-re0> show lldp neighbors interface xe-3/0/0.0
LLDP Neighbor Information:
Local Information:
Index: 37 Time to live: 20 Time mark: Mon May 10 14:48:22 2010 Age: 2 secs
Local Interface    : xe-3/0/0.0
Local Port ID      : 519

Neighbour Information:
Chassis type       : Mac address
Chassis ID         : 00:26:88:87:e3:f0
Port type          : Locally assigned
Port ID            : 206
Port description   : xe-3/0/1.0
System name        : J08E-re0

System Description : Juniper Networks, Inc. IBM 4274-E08 J08E , version 10.1R1.8
Build date: 2010-02-12 17:35:26 UTC

System capabilities
        Supported: Bridge Router
        Enabled  : Bridge Router

Management Info
        Type     : IPv4
        Address  : 10.1.1.30
        Port ID           : 34
        Subtype           : 1
        Interface Subtype : 2
        OID               : 1.3.6.1.2.1.31.1.1.1.1.34

Organization Info
       OUI      : 0.18.15
       Subtype  : 1
       Index    : 1
       Info     : 0000010000

Organization Info
       OUI      : 0.18.15
       Subtype  : 3
       Index    : 2
       Info     : 0100000000

Organization Info
       OUI      : 0.18.15
       Subtype  : 4
       Index    : 3
       Info     : 05EA
```

```
Organization Info
       OUI      : 0.18.15
       Subtype  : 3
       Index    : 4
       Info     : 00C80753657276657273

Organization Info
       OUI      : 0.18.15
       Subtype  : 3
       Index    : 5
       Info     : 0064055573657273

{master}
ibm@J08E-re0> show lldp local-information

LLDP Local Information details

Chassis ID   : 00:26:88:87:e3:f0
System name  : J08E-re0
System descr : Juniper Networks, Inc. IBM 4274-E08 J08E , version 10.1R1.8
               Build date: 2010-02-12 17:35:26 UTC

System Capabilities
    Supported       : Bridge Router
    Enabled         : Bridge Router

Management Information
    Port Name       : me0.0
    Port ID         : 34
    Port ID Subtype : local(7)
    Port Subtype    : ifIndex(2)

Interface name    Interface ID    Interface description    Status      Tunneling
me0.0             34              me0.0                    Up          Disabled
xe-3/0/0.0        519             xe-3/0/0.0               Up          Disabled
xe-3/0/1.0        206             xe-3/0/1.0               Up          Disabled
ge-1/0/15.0       520             -                        Down        Disabled
ge-1/0/11.0       513             -                        Down        Disabled
ge-1/0/12.0       514             -                        Down        Disabled
ge-1/0/13.0       517             -                        Down        Disabled
ge-1/0/14.0       518             -                        Down        Disabled

{master}
ibm@J08E-re0> show lldp statistics
Interface  Received  Unknown TLVs  With Errors  Discarded TLVs  Transmitted
Untransmitted
me0.0      0         0            0            0               14272        0
xe-3/0/0.0 4830      0            0            0               4831         0
xe-3/0/1.0 4830      0            0            0               4831         0

{master}
ibm@J08E-re0>
```

9.7 More information

All of the following documents are available on the IBM support site at this address:

http://www.ibm.com/systems/support/supportsite.wss/brandmain?brandind=5375876

For additional information and step-by-step installation procedures, see the following documentation as appropriate for your switch:

- *IBM Ethernet Switch J48E Complete Hardware Guide,* GA32-0663
- *IBM Ethernet Switch J48E Quick Start,* GA32-0664
- *IBM Ethernet Switch J08E Complete Hardware Guide,* GA32-0665
- *IBM Ethernet Switch J08E Quick Start,* GA32-0666
- *IBM Ethernet Switch J16E Complete Hardware Guide,* GA32-0667
- *IBM Ethernet Switch J16E Quick Start,* GA32-0668

For more information about the IBM j-type Ethernet routers the related hardware, see the following documentation:

- *IBM j-type m-series Ethernet Router Dense Port Concentrators (DPC) Guide,* GA32-0661
- *IBM j-type m-series Ethernet Routing Engine Installation Instructions,* GA32-0682
- *IBM j-type m-series Ethernet Services PIC Guide,* GA32-0662
- *IBM Ethernet Router J02M Hardware Guide,* GA32-0655
- *IBM Ethernet Router J02M Quick Start,* GA32-0656
- *IBM Ethernet Router J06M Hardware Guide,* GA32-0657
- *IBM Ethernet Router J06M Quick Start,* GA32-0658
- *IBM Ethernet Router J11M Hardware Guide,* GA32-0659
- *IBM Ethernet Router J11M Quick Start,* GA32-0660
- *JUNOS Software IBM j-type m-series Ethernet Routers Solutions Guide,* GA32-0683

For more information about the Junos software, see the following documentation:

- *Juniper Web Device Manager for IBM j-type Ethernet Switches and Routers Interface User Guide,* GA32-0688
- *JUNOS Software Access Privilege Configuration Guide,* GA32-0696
- *JUNOS Software Broadband Subscriber Management Solutions Guide,* GA32-0709
- *JUNOS Software Class of Service Configuration Guide,* GA32-0738
- *JUNOS Software CLI User Guide,* GA32-0697
- *JUNOS Software Configuration and Diagnostic Automation Guide,* GA32-0679
- *JUNOS Software Ethernet Routing Engine Media Upgrade Kit,* GA32-0681
- *JUNOS Software Feature Guide,* GA32-0739
- *JUNOS Software Hierarchy and RFC Reference,* GA32-0712
- *JUNOS Software High Availability Configuration Guide,* GA32-0670
- *JUNOS Software IBM j-type m-series Ethernet Routers Layer 2 Configuration Guide,* GA32-0708
- *JUNOS Software Installation and Upgrade Guide,* GA32-0695
- *JUNOS Software Interfaces Command Reference,* GA32-0672

- *JUNOS Software JUNOScript API Guide,* GA32-0674
- *JUNOS Software MPLS Applications Configuration Guide,* GA32-0702
- *JUNOS Software Multicast Protocols Configuration Guide,* GA32-0703
- *JUNOS Software NETCONF API Guide,* GA32-0678
- *JUNOS Software Network Interfaces Configuration Guide,* GA32-0706
- *JUNOS Software Network Management Configuration Guide,* GA32-0698
- *JUNOS Software Policy Framework Configuration Guide,* GA32-0704
- *JUNOS Software Routing Protocols and Policies Command Reference,* GA32-0673
- *JUNOS Software Services Interfaces Configuration Guide,* GA32-0707
- *JUNOS Software Subscriber Access Configuration Guide,* GA32-0711
- *JUNOS Software System Basics and Services Command Reference,* GA32-0671
- *JUNOS Software System Log Messages Reference,* GA32-0675
- *JUNOS Software VPNs Configuration Guide,* GA32-0705
- *JUNOScope Software User Guide,* GA32-0670

10

Layer 3: Routing configuration

This chapter explains how to configure Layer 3 capabilities on IBM j-type Ethernet switches and routers. It presents some common configuration scenarios for data center implementations. For complete and advanced information, see the additional documents listed in 10.7, “More information” on page 330.

This chapter includes the following topics:

- Network topology for Layer 3 configuration
- IP routing
- Routing protocols
- Static routes
- Multicast
- Virtual Router Redundancy Protocol

Before reading this chapter, you must have a basic understanding of networking concepts. If necessary, see Chapter 1, “Fundamentals of Ethernet networking” on page 1, and *IBM j-type Data Center Networking Introduction*, SG24-7820, for a short introduction.

© Copyright IBM Corp. 2011. All rights reserved.

10.1 Network topology for Layer 3 configuration

Figure 10-1 illustrates the physical connections of the lab equipment used for demonstrating the examples from this chapter. Only the connections that are relevant to this chapter are presented. Three units of IBM Ethernet Switch J48Es are interconnected to build a single management switch called the *Virtual Chassis switch*.

Figure 10-1 Physical network topology for Layer 3 configurations

Figure 10-2 shows the IP networks that are used for demonstrating the examples in this chapter. We only present the IP networks that are relevant to this chapter. In our lab, we created two pairs of equipment, one pair made of J48E-VC switches and J04E switch, and another pair made of J11M and J02M routers. The two pairs have similar IP addresses to test same configuration on both platforms.

Because the features demonstrated here are same on both pairs of equipment, we selected to show only one common configuration. For OSPF, static routes, VRRP, Multicast, and RIP, we present the configuration from the J04E switch. For Border Gateway Protocol (BGP), we present the configuration from the J02M router.

Figure 10-2 Logical network topology for Layer 3 configurations

10.2 IP routing

This section provides the fundamentals of IP routing on IBM j-type Ethernet switches and routers.

10.2.1 Routing protocol databases

Each Interior Gateway Protocol (IGP) routing protocol maintains a database of the routing information it has learned from other routers that are running the same protocol. It uses this information as defined and required by the protocol. IS-IS and OSPF use the routing information they receive to maintain link-state databases. They use these databases to determine which adjacent neighbors are operational and to construct network topology maps.

To determine the best route or routes (if there are multiple equal-cost routes) to reach each destination and install these routes into the Junos software routing table, use the following algorithms are used:

- IS-IS and OSPF use the Dijkstra algorithm.
- RIP and RIPng use the Bellman-Ford algorithm.

When you configure a protocol on an interface, you must also configure a protocol family on that interface.

10.2.2 Junos routing tables

The Junos software routing table is used by the routing protocol process to maintain its database of routing information. In this table, the routing protocol process stores statically configured routes, directly connected interfaces (also called *direct routes* or *interface routes*), and all routing information learned from all routing protocols. The routing protocol process uses this collected routing information to select the active route to each destination, which is the route that is used to forward packets to that destination.

By default, the Junos software maintains three routing tables: one for unicast routes, one for multicast routes, and one for Multiprotocol Label Switching (MPLS). You can configure additional routing tables to support situations where you must separate a particular group of routes or where you need greater flexibility in manipulating routing information. In general, most operations can be performed without resorting to the complexity of additional routing tables. However, creating additional routing tables has several specific uses. For example, you can import interface routes into more than one routing table, apply different routing policies when exporting the same route to different peers, and provide greater flexibility with incongruent multicast topologies.

Each routing table is identified by a name, which consists of the protocol family followed by a period and a small, non-negative integer. The protocol family can be inet (Internet), iso (ISO), or mpls (MPLS). The following names are reserved for the default routing tables maintained by the Junos software:

inet.0	Default IP version 4 (IPv4) unicast routing table.
inet6.0	Default IP version 6 (IPv6) unicast routing table.
instance-name.inet.0	Unicast routing table for a particular routing instance.
inet.1	Multicast forwarding cache.
inet.2	Unicast routes used for multicast reverse path forwarding (RPF) lookup.
inet.3	MPLS routing table for path information.
mpls.0	MPLS routing table for label-switched path (LSP) next hops.

Routing tables: For clarity, this chapter contains general discussions about routing tables as though there were only one table. However, when it is necessary to distinguish among the routing tables, their names are explicitly used.

10.2.3 Forwarding tables

The Junos software installs all active routes from the routing table into the forwarding table. The active routes are used to forward packets to their destinations. The Junos kernel maintains a master copy of the forwarding table. It copies the forwarding table to the Packet

Forwarding Engine (PFE), which is the part of the router that is responsible for forwarding packets.

10.2.4 Synchronizing the routing and forwarding tables

The Junos routing protocol process is responsible for synchronizing the routing information between the routing and forwarding tables. To synchronize this information, the routing protocol process calculates the active routes from all the routes in the routing table and installs them into the forwarding table. The routing protocol process then copies the forwarding table to the PFE of the router, which is the part of the router that forwards packets.

10.2.5 Route preferences overview

For unicast routes, the Junos routing protocol process uses the information in its routing table, and the properties set in the configuration file, to select an active route for each destination. The Junos software might know of many routes to a destination. However, the active route is the preferred route to that destination. It is the route that is installed in the forwarding table and used when routing packets.

The routing protocol process generally determines the active route by selecting the route with the lowest preference value. The preference value is an arbitrary value in the range 0–4,294,967,295 ($2^{32} - 1$) that the software uses to rank routes received from different protocols, interfaces, or remote systems.

The preference value is used to select routes to destinations in external autonomous systems or routing domains. It has no effect on the selection of routes within an AS (autonomous systems, that is, within an IGP). Routes within an AS are selected by the IGP and are based on that metric or cost value of the protocol.

10.2.6 Multiple active routes

The IGPs compute equal-cost multipath next hops, and internal BGP (IBGP) picks up these next hops. When multiple, equal-cost next hops are associated with a route, the routing protocol process installs only one of the next hops in the forwarding path with each route. It randomly selects which hop to install next. For example, if there are 3 equal-cost paths to an exit routing device and 900 routes leaving through that routing device, about 300 routes are pointing to each path. This mechanism provides load distribution among the paths while maintaining packet ordering per destination.

10.2.7 Determining the active route

For each prefix in the routing table, the routing protocol process selects a single best path, called the *active route*. To determine the active route, you select the path with the lowest preference value (routing protocol process preference). Routes that are not eligible to be used for forwarding have a preference of –1 and are never selected. Examples of such routes include routes that were rejected by the routing policy or where a next hop is inaccessible.

10.2.8 Default route preference values

The Junos software routing protocol process assigns a default preference value to each route that the routing table receives. The default value depends on the source of the route. The preference value is a value of 0–4,294,967,295 ($2^{32} – 1$), with a lower value indicating a more preferred route. Table 10-1 lists the default preference values for Junos software.

Table 10-1 Default route preference values for Junos software

How a route is learned	Default preference
Directly connected network	0
System routes	4
Static and static LSPs	5
RSVP-signaled LSPs	7
LDP-signaled LSPs	9
OSPF internal route	10
IS-IS Level 1 internal route	15
IS-IS Level 2 internal route	18
Redirects	30
Kernel	40
SNMP	50
Router discovery	55
RIP	100
RIPng	100
Aggregate	130
OSPF AS external routes	150
IS-IS Level 1 external route	160
IS-IS Level 2 external route	165
BGP	170

10.3 Routing protocols

This section presents a brief overview of the RIP, OSPF, and BGP routing protocols. It also explains how to configure these routing protocols on IBM j-series Ethernet Switches and Routers.

10.3.1 RIP

RIP is an IGP that uses a distance-vector algorithm to determine the best route to a destination, using the hop count as the metric. The RIP IGP uses the Bellman-Ford, or distance-vector algorithm, to determine the best route to a destination. RIP uses the hop count as the metric. With RIP, hosts and routers can exchange information for computing

routes through an IP-based network. RIP is intended to be used as an IGP in reasonably homogeneous networks of moderate size.

The Junos software supports RIP versions 1 and 2. RIP version 1 packets contain the minimal information necessary to route packets through a network. However, this version of RIP does not support authentication or subnetting.

RIP uses User Datagram Protocol (UDP) port 520.

RIP has the following architectural limitations:

- The longest network path cannot exceed 15 hops, assuming that each network, or hop, has a cost of 1.
- RIP depends on counting to infinity to resolve certain unusual situations. When the network consists of several hundred routers, and when a routing loop has formed, the amount of time and network bandwidth required to resolve a next hop might be great.
- RIP uses only a fixed metric to select a route. Other IGPs use additional parameters, such as measured delay, reliability, and load.

RIP standards

RIP is defined in the following documents:

- RFC 1058, Routing Information Protocol
- RFC 2082, RIP-2 MD-5 Authentication
- RFC 2453, RIP Version 2

To access Internet Requests for Comments (RFCs) and drafts, go to the Internet Engineering Task Force (IETF) website at:

http://www.ietf.org

RIP configuration

On IBM j-type Ethernet switches and routers, RIP is disabled by default. To have a router exchange routes with other routers, you must configure RIP groups and neighbors. RIP routes received from routers that are not configured as RIP neighbors are ignored. Likewise, RIP routes are advertised only to routers that are configured as RIP neighbors, with an appropriate RIP export policy applied.

Minimum RIP configuration

For a routing device to accept RIP routes, you must include at least the **rip**, **group**, and **neighbor** statements. Routes received from routing devices that are not configured as neighbors are ignored. This minimum configuration defines one neighbor. Include one neighbor statement for each interface on which you want to receive routes. The local routing device imports all routes by default from this neighbor and does not advertise routes. The routing device can receive both version 1 and version 2 update messages, with 25 route entries per message.

To configure RIP on an interface, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip group rip_group neighbor interface_name
```

Configuring authentication for the RIP

You can configure the router to authenticate RIP route queries. By default, authentication is disabled. You can use the following authentication methods:

- *Simple authentication* uses a text password that is included in the transmitted packet. The receiving router uses an authentication key (password) to verify the packet.
- *Message-digest algorithm 5 (MD5) authentication* creates an encoded checksum that is included in the transmitted packet. The receiving router uses an authentication key (password) to verify the MD5 checksum of the packet.

To configure simple authentication for RIP, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip group rip_group neighbor interface_name
authentication-type simple authentication-key key
```

To configure MD5 authentication for RIP, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip group rip_group neighbor interface_name
authentication-type md5 authentication-key key
```

Configuring the metric value added to imported RIP routes

By default, RIP imports routes from the neighbors that are configured with the neighbor statement. These routes include routes that are learned from RIP and from other protocols. By default, the current route metric of routes that RIP imports from its neighbors is incremented by 1.

To change the default metric to be added to incoming routes, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip group rip_group neighbor interface_name metric-in
metric
```

where metric can be a value in the range 1–15.

Applying policies to routes exported by RIP

By default, RIP does not export routes that it has learned to its neighbors. To enable RIP to export routes, apply one or more export policies. To apply export policies and to filter routes that are being exported from the local routing device to its neighbors, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip group rip_group export export_policy
```

You can define one or more export policies. If no routes match the policies, the local routing device does not export any routes to its neighbors. Export policies override any metric values determined through calculations involving the values that are configured with the **metric-in** and **metric-out** statements.

To create a simple export policy with the name *export_policy*, use the following commands:

```
{master}[edit]
ibm@J08E-re0# set policy-options policy-statement export_policy from protocol
[direct|static|ospf|bgp]
{master}[edit]
ibm@J08E-re0# set policy-options policy-statement export_policy then accept
```

For more information about policies and their options, see *JUNOS Software Policy Framework Configuration Guide,* GA32-0704.

Configuring the metric for routes exported by RIP

If you included the **export** statement, RIP exports routes it has learned to the neighbors that are configured by including the neighbor statement.

The metric associated with a RIP route (unless modified by an export policy) is the normal RIP metric. For example, a RIP route with a metric of 5 learned from a neighbor configured with a metric-in value of 2 is advertised with a combined metric of 7 when advertised to RIP neighbors in the same group. However, if this route was learned from a RIP neighbor in a different group or from a different protocol, the route is advertised with the metric value configured for that group with the **metric-out** statement. The default value for the metric-out statement is 1.

The metric for a route can be modified with an export policy. That metric is seen when the route is exported to the next hop.

To increase the metric for routes advertised outside a group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip group rip_group metric-out metric
```

where metric can be a value in the range 1–15.

Configuring RIP update messages

You can configure whether the RIP update messages conform to RIP version 1 only, to RIP version 2 only, or to both versions. You can also disable the sending or receiving of update messages.

To configure the sending of update messages at a specific neighbor level, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip group rip_group neighbor interface_name send
(broadcast|multicast|none|version-1)
```

To configure the sending of update messages at the protocol level, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip send (broadcast|multicast|none|version-1)
```

The configuration at the neighbor level takes precedence over the one at protocol level.

To configure the receiving of update messages at a specific neighbor level, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip group rip_group neighbor interface_name receive
(none|version-1|version-2|both)
```

To configure the receiving of update messages at protocol level, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip receive (none|version-1|version-2|both)
```

The configuration at neighbor level takes precedence over the one at protocol level.

Configuring the default preference value for RIP

By default, the Junos software assigns a preference of 100 to routes that originate from RIP. When the Junos software determines a route's preference to become the active route, the software selects the route with the lowest preference and installs this route in the forwarding table.

To modify the default RIP preference value, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip group rip_group preference preference
```

preference can be a value in the range 0–4,294,967,295 ($2^{32} - 1$).

Configuring RIP timers

You can configure various timers for RIP. RIP routes expire when a route timeout limit is met or a route metric reaches infinity, and the route is no longer valid. However, the expired route is retained in the routing table for a time period so that neighbors can be notified that the route has been dropped. This time period is set by configuring the hold-down timer. Upon expiration of the hold-down timer, the route is removed from the routing table.

To change the hold-down timer for RIP, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip holddown timer
```

where *timer* is 10–180 seconds. The default value is 120 seconds.

To change the route-timeout timer for RIP, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip route-timeout timer
```

where *timer* is 60–360 seconds. The default value is 180 seconds.

To change the update-interval timer for RIP, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols rip update-interval timer
```

where *timer* is 10–60 seconds. The default value is 30 seconds.

Verifying RIP operations

To verify the operation of RIP, use the following commands:

```
{master}
ibm@J08E-re0> show rip statistics

{master}
ibm@J08E-re0> show rip neighbor

{master}
ibm@J08E-re0> show route protocol rip
```

Example 10-1 shows the configuration of three RIP groups with the following characteristics:

- RIP group 1
  - Neighbor interface of ge-1/0/1.0
  - No authentication
  - Standard timers
- RIP group 2
  - Neighbor interface of ge-1/0/2.0
  - Send RIP multicast updates
  - Receive only RIP version 2
  - Authentication simple with "key-group-2"
  - The addition of 2 to the metric of the incoming routes
- RIP group 3
  - Neighbor interface of ge-1/0/3.0
  - Receive RIP version 1 and 2
  - Send RIP multicast updates
  - Authentication MD5 with "key-group-3"
  - The addition of 3 to the metric of the incoming routes
  - Preference set to 101
  - RIP update interval set to 10 seconds
  - Metric of exported routes set to 3
  - Export of all direct routes to RIP neighbors

Example 10-1 RIP configuration

```
{master}[edit]
ibm@J08E-re0# edit protocols rip

{master}[edit protocols rip]
ibm@J08E-re0# set group group-1 neighbor ge-1/0/1.0

{master}[edit protocols rip]
ibm@J08E-re0# set group group-2 neighbor ge-1/0/2.0

{master}[edit protocols rip]
ibm@J08E-re0# set group group-2 neighbor ge-1/0/2.0 send multicast

{master}[edit protocols rip]
ibm@J08E-re0# set group group-2 neighbor ge-1/0/2.0 receive version-2

{master}[edit protocols rip]
ibm@J08E-re0# set group group-2 neighbor ge-1/0/2.0 authentication-type simple
authentication-key key-group-2

{master}[edit protocols rip]
ibm@J08E-re0# set group group-2 neighbor ge-1/0/2.0 metric-in 2

{master}[edit protocols rip]
ibm@J08E-re0# set group group-3 neighbor ge-1/0/3.0

{master}[edit protocols rip]
ibm@J08E-re0# set group group-3 neighbor ge-1/0/3.0 send multicast

{master}[edit protocols rip]
```

```
ibm@J08E-re0# set group group-3 neighbor ge-1/0/3.0 receive both

{master}[edit protocols rip]
ibm@J08E-re0# set group group-3 neighbor ge-1/0/3.0 authentication-type md5
authentication-key key-group-3

{master}[edit protocols rip]
ibm@J08E-re0# set group group-3 neighbor ge-1/0/3.0 metric-in 3

{master}[edit protocols rip]
ibm@J08E-re0# set group group-3 preference 101

{master}[edit protocols rip]
ibm@J08E-re0# set group group-3 neighbor ge-1/0/3.0 update-interval 10

{master}[edit protocols rip]
ibm@J08E-re0# set group group-3 metric-out 2

{master}[edit protocols rip]
ibm@J08E-re0# set group group-3 export conn_to_rip

{master}[edit protocols rip]
ibm@J08E-re0# top

{master}[edit]
ibm@J08E-re0# set policy-options policy-statement conn_to_rip from protocol direct

{master}[edit]
ibm@J08E-re0# set policy-options policy-statement conn_to_rip then accept

{master}[edit]
ibm@J08E-re0# show protocols rip
group group-1 {
    neighbor ge-1/0/1.0;
}
group group-2 {
    neighbor ge-1/0/2.0 {
        metric-in 2;
        send multicast;
        receive version-2;
        authentication-type simple;
        authentication-key "$9$MM4XNb4oGDkPsYGiq.F3Ecyl87NdsaZUdV"; ## SECRET-DATA
    }
}
group group-3 {
    preference 101;
    metric-out 2;
    export conn_to_rip;
    neighbor ge-1/0/3.0 {
        update-interval 10;
        metric-in 3;
        send multicast;
        receive both;
        authentication-type md5;
        authentication-key "$9$xdAdwgGUHqfzoaHm5T9CrevWNbwYoDik2g"; ## SECRET-DATA
```

```
    }
}

{master}[edit]
ibm@J08E-re0# show policy-options
policy-statement conn_to_rip {
    from protocol direct;
    then accept;
}

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# run show rip neighbor
                        Source          Destination     Send   Receive   In
Neighbor          State  Address         Address         Mode   Mode     Met
--------          -----  -------         -----------     ----   -------  ---
ge-1/0/1.0           Up 172.16.1.1      224.0.0.9       mcast  both       1
ge-1/0/2.0           Up 172.16.2.1      224.0.0.9       mcast  v2 only    2
ge-1/0/3.0           Up 172.16.3.1      224.0.0.9       mcast  both       3

{master}[edit]
ibm@J08E-re0# run show route protocol rip

inet.0: 30 destinations, 32 routes (30 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.10.0/24    *[RIP/100] 00:05:33, metric 2, tag 1 <--- Preference is 100
                    > to 172.16.1.3 via ge-1/0/1.0
192.168.11.0/24    *[RIP/100] 00:05:33, metric 2, tag 1 <--------- Metric is 2
                    > to 172.16.1.3 via ge-1/0/1.0
192.168.12.0/24    *[RIP/100] 00:05:33, metric 2, tag 1
                    > to 172.16.1.3 via ge-1/0/1.0
192.168.13.0/24    *[RIP/100] 00:05:33, metric 2, tag 1
                    > to 172.16.1.3 via ge-1/0/1.0
192.168.20.0/24    *[RIP/100] 00:20:21, metric 3, tag 2 <--------- Metric is 3
                    > to 172.16.2.3 via ge-1/0/2.0
192.168.21.0/24    *[RIP/100] 00:20:21, metric 3, tag 2
                    > to 172.16.2.3 via ge-1/0/2.0
192.168.22.0/24    *[RIP/100] 00:20:21, metric 3, tag 2
                    > to 172.16.2.3 via ge-1/0/2.0
192.168.23.0/24    *[RIP/100] 00:20:21, metric 3, tag 2
                    > to 172.16.2.3 via ge-1/0/2.0
192.168.30.0/24    *[RIP/101] 00:27:15, metric 4, tag 3 <---- Preference is 101
                    > to 172.16.3.3 via ge-1/0/3.0
192.168.31.0/24    *[RIP/101] 00:27:15, metric 4, tag 3 <--------- Metric is 4
                    > to 172.16.3.3 via ge-1/0/3.0
192.168.32.0/24    *[RIP/101] 00:27:15, metric 4, tag 3
```

```
                    > to 172.16.3.3 via ge-1/0/3.0
192.168.33.0/24    *[RIP/101] 00:27:15, metric 4, tag 3
                    > to 172.16.3.3 via ge-1/0/3.0
224.0.0.9/32       *[RIP/100] 00:11:45, metric 1
                      MultiRecv

inet6.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

{master}[edit]
ibm@J08E-re0# run show route 192.168.10.0/24 extensive

inet.0: 30 destinations, 32 routes (30 active, 0 holddown, 0 hidden)
192.168.10.0/24 (1 entry, 1 announced)
TSI:
KRT in-kernel 192.168.10.0/24 -> {172.16.1.3}
        *RIP    Preference: 100
                Next hop type: Router, Next hop index: 1345
                Next-hop reference count: 8
                Next hop: 172.16.1.3 via ge-1/0/1.0, selected
                State: <Active Int>
                Age: 3:42 Metric: 2 Tag: 1
                Task: RIPv2
                Announcement bits (1): 0-KRT
                AS path: I
      Route learned from 172.16.1.3 expires in 167 seconds

{master}[edit]
ibm@J08E-re0# run show rip statistics
RIPv2 info: port 520; holddown 120s.
    rts learned  rts held down  rqsts dropped  resps dropped
             12              0              0              0

ge-1/0/1.0:  4 routes learned; 0 routes advertised; timeout 180s; update interval
30s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                        0            0            0
Triggered Updates Sent              0            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received              7            7            2
RIPv2 Bad Route Entries             0            0            0
RIPv2 Updates Ignored               0            0            0
Authentication Failures             0            0            0
RIP Requests Received               0            0            0
RIP Requests Ignored                0            0            0

ge-1/0/2.0:  4 routes learned; 0 routes advertised; timeout 180s; update interval
30s
Counter                         Total   Last 5 min  Last minute
-------                   -----------  -----------  -----------
Updates Sent                        0            0            0
```

```
Triggered Updates Sent              0            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received              7            7            3
RIPv2 Bad Route Entries             0            0            0
RIPv2 Updates Ignored               0            0            0
Authentication Failures             0            0            0
RIP Requests Received               0            0            0
RIP Requests Ignored                0            0            0

ge-1/0/3.0:  4 routes learned; 4 routes advertised; timeout 180s; update interval
10s
Counter                         Total   Last 5 min  Last minute
-------                     -----------  -----------  -----------
Updates Sent                       19           19            6 <--- Update sent
Triggered Updates Sent              0            0            0
Responses Sent                      0            0            0
Bad Messages                        0            0            0
RIPv1 Updates Received              0            0            0
RIPv1 Bad Route Entries             0            0            0
RIPv1 Updates Ignored               0            0            0
RIPv2 Updates Received             20           19            6
RIPv2 Bad Route Entries             0            0            0
RIPv2 Updates Ignored               0            0            0
Authentication Failures             0            0            0
RIP Requests Received               0            0            0
RIP Requests Ignored                0            0            0

{master}[edit]
ibm@J08E-re0#
```

10.3.2 OSPF

OSPF is an IGP that routes packets within a single AS. OSPF uses link-state information to make routing decisions, making route calculations using the Shortest Path First (SPF) algorithm (also known as the Dijkstra algorithm). Each router running OSPF floods link-state advertisements throughout the AS that contain information about that router's attached interfaces and routing metrics. Each router takes the information in these link-state advertisements and creates a complete routing table for the network.

The Junos software supports OSPF version 2, including virtual links, stub areas, and authentication. The Junos software does not support type-of-service (ToS) routing. OSPF was designed for the Transmission Control Protocol/IP (TCP/IP) environment. As a result, it explicitly supports IP subnetting and the tagging of externally derived routing information. OSPF also provides for the authentication of routing updates.

OSPF routes IP packets based solely on the destination IP address contained in the IP packet header. OSPF quickly detects topological changes, such as when router interfaces become unavailable. It also calculates new loop-free routes quickly and with a minimum of routing overhead traffic.

Each interface running OSPF is assigned a cost, which is a unitless number based on factors such as throughput, round-trip time, and reliability. These factors are used to determine how easy or difficult it is to reach a destination. If two or more routes to a destination have the same cost, OSPF distributes traffic equally among the routes. This process is called *load balancing*.

Each router maintains a database that describes the topology of the AS. Each OSPF router has an identical topological database so that all routers in the area have a consistent view of the network. All routers maintain summarized topologies of other areas within an AS. Each router distributes information about its local state by flooding link-state advertisements throughout the AS. When the AS topology changes, OSPF ensures that the contents of the topological databases of all routers converge quickly.

All OSPF protocol exchanges can be authenticated, meaning that only trusted routers can participate in the routing of the AS. Various authentication schemes can be used. A single authentication scheme is configured for each area, which enables some areas to use stricter authentication than others.

Externally derived routing data (for example, routes learned from BGP) is passed transparently throughout the AS. This externally derived data is kept separate from the OSPF link-state data. Each external route can be tagged by the advertising router, enabling the passing of additional information between routers on the boundaries of the AS.

OSPF routing algorithm

OSPF uses the SPF algorithm to determine the route to reach each destination. All routers in an area run this algorithm in parallel, storing the results in their individual topological databases. Routers with interfaces to multiple areas run multiple copies of the algorithm. This section provides a brief summary of how the SPF algorithm works.

When a router starts, it initializes OSPF and waits for indications from lower-level protocols that the router interfaces are functional. The router then uses the OSPF Hello protocol to acquire neighbors by sending Hello packets to its neighbors and receiving their Hello packets.

On broadcast or nonbroadcast multi-access networks (physical networks that support the attachment of more than two routers), the OSPF Hello protocol elects a designated router for the network. This router is responsible for sending link-state advertisements that describe the network, which reduces the amount of network traffic and the size of the routers' topological databases.

The router then attempts to form adjacencies with some of its newly acquired neighbors. (On multi-access networks, only the designated router and backup designated router form adjacencies with other routers.) Adjacencies determine the distribution of routing protocol packets. Routing protocol packets are sent and received only on adjacencies, and topological database updates are sent only along adjacencies. When adjacencies have been established, pairs of adjacent routers synchronize their topological databases.

A router sends Link State Advertisement (LSA) packets to advertise its state periodically and when the state of the router changes. These packets include information about the adjacencies of the router, which allows detection of nonoperational routers.

By using a reliable algorithm, the router floods LSAs throughout the area, which ensures that all routers in an area have the same topological database. Each router uses the information in its topological database to calculate a shortest-path tree, with itself as the root. The router then uses this tree to route network traffic.

The description of the SPF algorithm up to this point has explained how the algorithm works within a single area (intra-area routing). For internal routers to route to destinations outside the area (inter-area routing), the area border routers must inject additional routing information into the area. Because the area border routers are connected to the backbone, they have access to complete topological data about the backbone. They use this information to calculate paths to all destinations outside its area and then advertise these paths to the internal routers of the area.

AS boundary routers flood information about external autonomous systems throughout the AS, except to stub areas. Area border routers are responsible for advertising the paths to all AS boundary routers.

OSPF areas and OSPF routers

In OSPF, a single AS can be divided into smaller groups called *areas*. This method reduces the number of link-state advertisements and other OSPF overhead traffic sent in the network. It also reduces the size of the topological database that each router must maintain.

An *area* is a set of networks and hosts within an AS that have been administratively grouped together. Configure an area as a collection of contiguous IP subnetted networks. Routers that are wholly within an area are called *internal routers*. All interfaces on internal routers are directly connected to networks within the area.

The topology of an area is hidden from the rest of the AS, significantly reducing routing traffic in the AS. Also, routing within the area is determined only by the topology of the area, providing the area with protection from bad routing data. All routers within an area have identical topological databases.

Area border routers

Routers that belong to more than one area are called *area border routers*. They maintain a separate topological database for each area to which they are connected.

Backbone areas

An OSPF backbone area consists of all networks in area ID 0.0.0.0, their attached routers, and all area border routers. The backbone itself does not have any area border routers. The backbone distributes routing information between areas. The backbone is another area. Therefore, the terminology and rules of areas apply. A router that is directly connected to the backbone is an internal router on the backbone, and the backbone's topology is hidden from the other areas in the AS.

The routers that make up the backbone must be physically contiguous. If they are not, you must configure virtual links to create the appearance of backbone connectivity. You can create virtual links between any two area border routers that have an interface to a common nonbackbone area. OSPF treats two routers joined by a virtual link as though they were connected to an unnumbered point-to-point network.

AS boundary routers

Routers that exchange routing information with routers in other autonomous systems are called *AS boundary routers*. They advertise externally learned routes throughout the AS. Any router in the AS, which can be an internal router, an area border router, or a backbone router, can be an AS boundary router.

Every router within the AS knows the path to the AS boundary routers.

Stub areas

Stub areas are areas through which or into which AS external advertisements are not flooded. You might want to create stub areas when much of the topological database consists of AS external advertisements. This method reduces the size of the topological databases and, therefore, the amount of memory required on the internal routers in the stub area.

When an area border router is configured for a stub area, the router automatically advertises a default route in place of the external routes that are not advertised within the stub area. This way routers in the stub area can reach destinations outside the area.

The following restrictions apply to stub areas:

- You cannot create a virtual link through a stub area
- A stub area cannot contain an AS boundary router.

Not-so-stubby areas

An OSPF stub area has no external routes in it. Therefore, you cannot redistribute it from another protocol into a stub area. In a not-so-stubby area (NSSA), external routes can be flooded within the area. These routes are then leaked into other areas. However, external routes from other areas still do not enter the NSSA.

Transit areas

Transit areas are used to pass traffic from one adjacent area to the backbone (or to another area if the backbone is more than two hops away from an area). The traffic does not originate in, nor is it destined for, the transit area.

Overview of OSPF external metrics

When OSPF exports route information from external autonomous systems, it includes a cost, or external metric, in the route. Two types of external metrics are possible: Type 1 and Type 2.

Type 1 external metrics are equivalent to the link-state metric, which is the cost of the route used in the internal AS. Type 2 external metrics are greater than the cost of any path internal to the AS.

Overview of the OSPF designated router

Each multi-access network has a designated router, which performs two main functions:

- Originate network link advertisements on behalf of the network.
- Establish adjacencies with all routing devices in the network, thus participating in the synchronizing of the link-state databases.

The OSPF Hello protocol elects a designated router for the network based on the priorities advertised by all the routing devices. In general, when an interface first becomes functional, it checks whether the network currently has a designated router. If the network has a designated router, the routing device accepts that designated router regardless of its own router priority. Otherwise, if the routing device has the highest priority in the network, it becomes the designated router. If router priorities tie, the routing device with the highest router ID (typically the IP address of the routing device) is selected as the designated router.

OSPF standards

OSPF and OSPFv3 are defined in the following documents:

- RFC 1793, Extending OSPF to Support Demand Circuits
- RFC 2328, OSPF Version 2
- RFC 2370, The OSPF Opaque LSA Option

- RFC 2740, OSPF for IPv6
- RFC 3101, The OSPF Not-So-Stubby Area (NSSA) Option
- RFC 3509, Alternative Implementations of OSPF Area Border Routers
- RFC 3623, OSPF Graceful Restart
- RFC 3630, Traffic Engineering (TE) Extensions to OSPF Version 2
- RFC 4203, OSPF Extensions in Support of Generalized Multiprotocol Label Switching (GMPLS) (only interface switching)
- RFC 4552, Authentication/Confidentiality for OSPFv3
- RFC 4576, Using a Link State Advertisement (LSA) Options Bit to Prevent Looping in BGP/MPLS Virtual Private Networks (VPNs)
- RFC 4577, OSPF as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)
- RFC 5185, OSPF Multi-Area Adjacency
- Internet draft `draft-ietf-katz-ward-bfd-00.txt`, Bidirectional Forwarding Detection (except the transmission of echo packets) (expires January 2005)
- Internet draft `draft-ietf-isis-igp-p2p-over-lan-03.txt`, Point-to-point operation over LAN in link-state routing protocols (expires February 2004)
- Internet draft `draft-ospf-alt-06.txt`, Support of address families in OSPFv3 (expires April 2008)

To access Internet RFCs and drafts, go to the IETF website at:

http://www.ietf.org

OSPF configuration

This section provides information about the configuration of the OSPF routing protocol on IBM j-type Ethernet switches and routers.

Configuring the OSPF backbone area

You must create a backbone area if your network consists of multiple areas. An area border router (ABR) must have at least one interface in the backbone area, or it must have a virtual link to a routing device in the backbone area. The backbone consists of all ABRs and all routing devices that are not included in any other area.

To configure a routing device to be in a backbone area, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area 0
```

Configuring the OSPF nonbackbone areas

Each OSPF area consists of routing devices configured with the same area number. The area number (*area_id*) can be any number except 0, which is reserved for the backbone area.

To configure a routing device to be in a nonbackbone area, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id
```

Configuring the OSPF stub areas

Stub areas are areas into which OSPF does not flood AS external advertisements. You might want to configure stub areas when much of the topological database consists of AS external advertisements. You migh also want to minimize the size of the topological databases on the

routing devices of an area. You cannot configure an area as being both a stub area and an NSSA.

To configure a stub area, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id stub
```

To inject a default route with a specified metric value into the area, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id stub default-metric metric
```

The default route matches any destination that is not explicitly reachable from within the area.

To have the stub areas not advertise summary routes into the stub area, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id stub no-summaries
```

In this case, only the default route is advertised, and only if you include the `default-metric` option. The default route injected into the NSSA is a Type 3 LSA.

Stub statement: You must include the stub statement when configuring all routing devices that are in the stub area.

Configuring the OSPF NSSA

An OSPF stub area has no external routes. Therefore, you cannot redistribute from another protocol into a stub area. WIth an NSSA, external routes can be flooded within the area. These routes are then leaked into other areas. However, external routes from other areas still do not enter the NSSA. You cannot configure an area to be both a stub area and an NSSA.

To configure an NSSA, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id nssa
```

By default, a default route is not advertised. To advertise a default route with the specified metric within the area, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id nssa default-lsa default-metric
metric
```

You can configure the `default-metric` option only on ABRs. The default route is a Type 3 LSA injected into the NSSA. To send the default route as a Type 7 LSA, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id nssa default-lsa type-7
```

To prevent an ABR from advertising summary routes into an NSSA, include the `no-summaries` statement:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id nssa no-summaries
```

If you include the `default-metric` option in addition to the `no-summaries` statement, only the default route is advertised.

To flood summary LSAs into the NSSA, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id nssa summaries
```

When summaries are configured (which is the default if the `no-summaries` statement is not specified), a Type 7 LSA is sent. To define the type of metric, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id nssa default-lsa metric-type (1|2)
```

To aggregate external routes learned within the area when a route is advertised to other areas, include one or more area-range statements as in the following example:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id nssa area-range area_range
```

Configuring OSPF on interfaces

To enable OSPF on the routing device, you must configure OSPF on at least one of the interfaces of the routing device. How you configure an interface depends on whether the interface is connected to a broadcast or point-to-point network, a point-to-multipoint network, or a nonbroadcast, multiaccess network. For the data center topology, we limit our configuration to broadcast media (Ethernet).

To configure a broadcast interface to be part of an OSPF area, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id interface interface
```

Configuring authentication for OSPFv2

All OSPFv2 protocol exchanges can be authenticated to guarantee that only trusted routers participate in the routing of the autonomous system. By default, OSPFv2 authentication is disabled. Junos software supports MD5, simple, and IPsec authentication.

> **IPsec authentication:** You can configure IPsec authentication with MD5 or simple authentication, but simple and MD5 authentication are mutually exclusive.

This section covers only the most common type of OSPF authentication, which is simple and MD5 authentication. For IPsec authentication, see the appropriate documents listed in 10.7, “More information” on page 330.

To configure OSPF simple authentication, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id interface interface authentication
simple-password password
```

To configure OSPF MD5 authentication, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id interface interface authentication
md5 key-id key md5_key
```

Summarizing the ranges of routes in OSPF link-state advertisements

Area border routers send summary link advertisements to describe the routes to other areas. You can minimize the number of these advertisements that are flooded. For example, you can configure the routing device to coalesce, or summarize, a range of IP addresses and send reachability information about these addresses in a single link-state advertisement.

To summarize a range of IP addresses in an OSPF area, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id area-range network/mask-length
```

To summarize multiple ranges, include the multiple `area-range` statements.

Configuring the metric value for OSPF interfaces

All OSPF interfaces have a cost, which is a routing metric that is used in the link-state calculation. Routes with lower total path metrics are preferred over routes with higher path metrics. When several equal-cost routes to a destination exist, traffic is distributed equally among them.

The cost of a route is described by a single dimensionless metric that is determined using the following formula:

```
cost = reference-bandwidth / bandwidth
```

You can modify the `reference-bandwidth` value. The bandwidth value refers to the actual bandwidth of the physical interface.

You can override the default behavior of using the reference bandwidth to calculate the metric cost of a route by configuring a specific metric value for any OSPF interface.

To modify the reference bandwidth, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf reference-bandwidth reference-bandwidth
```

The default value of the reference bandwidth is 100 Mbps (which you specify as 100,000,000). This bandwidth gives a metric of 1 for any interface with a physical bandwidth that is 100 Mbps or greater. For `reference-bandwidth`, you can configure a value of 9600–1,000,000,000,000 bits.

When you specify a metric for a specific OSPF interface, that value is used to determine the cost of routes advertised from that interface. To specify a metric for routes advertised from an interface, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id interface interface metric metric
```

For *`metric`*, you can specify a value of 1–65,535.

Configuring preference values for OSPF routes

Route preferences are used to select which route is installed in the forwarding table when several protocols calculate routes to the same destination. The route with the lowest preference value is selected. By default, internal OSPF routes have a preference value of 10, and external OSPF routes have a value of 150.

To change the preference values for internal routes, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf preference preference
```

To change the preference values for external routes, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf external-preference preference
```

The preference can be a value of 0–4,294,967,295 ($2^{32} – 1$).

Configuring OSPF timers

OSPF routing devices constantly track the status of their neighbors. They send and receive Hello packets that indicate whether the neighbor still is functioning, and send and receive link-state advertisement and acknowledgment packets. OSPF sends packets and expects to receive packets at specified intervals.

To modify how often the routing device sends Hello packets out of an interface, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id interface interface hello-interval
interval
```

By default, the routing device waits 5 seconds for an acknowledgment before retransmitting the link-state advertisement. To modify this interval, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id interface interface
retransmit-interval interval
```

To modify the router dead interval, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id interface interface dead-interval
interval
```

This interval must be the same for all routing devices on a shared network.

Configuring OSPF passive interface

OSPF uses all defined interfaces for sending and receiving OSPF packets. To advertise an interface in OSPF, but to prevent the sending and receiving of packets and establishing neighborship on it, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area area_id interface interface passive
```

OSPF verification

To verify the status of the OSPF, use the following commands:

```
{master}
ibm@J08E-re0> show ospf neighbor

{master}
ibm@J08E-re0> show ospf interface

{master}
ibm@J08E-re0> show ospf statistics

{master}
ibm@J08E-re0> show routes protocol ospf
```

Example 10-2 shows the configuration of three OSPF areas with the following characteristics:

- OSPF area 0
  - Interface ge-1/0/1.0
  - Hello interval 1 sec
  - Metric on ge-1/0/1.0 interface of 10
- OSPF area 1
  - Stub area
  - No summaries
  - Default route metric of 10
  - Authentication simple with "key-2"
- OSPF area 2
  - NSSA
  - Default route with type 2 and metric 20
  - Default route advertised as LSA type-7
  - Authentication md5, key 1 with "key-3"
  - Reference bandwidth set to 100g

Example 10-2 OSPF configuration

```
{master}[edit]
ibm@J08E-re0# set protocols ospf area 0 interface ge-1/0/1.0 metric 10

{master}[edit]
ibm@J08E-re0# set protocols ospf area 0 interface ge-1/0/1.0 hello-interval 1

{master}[edit]
ibm@J08E-re0# edit protocols ospf area 1

{master}[edit protocols ospf area 0.0.0.1]
ibm@J08E-re0# set stub no-summaries

{master}[edit protocols ospf area 0.0.0.1]
ibm@J08E-re0# set default-metric 10

{master}[edit protocols ospf area 0.0.0.1]
ibm@J08E-re0# set interface ge-1/0/2.0 authentication simple-password key-2

{master}[edit protocols ospf area 0.0.0.1]
ibm@J08E-re0# top edit protocols ospf area 2

{master}[edit protocols ospf area 0.0.0.2]
ibm@J08E-re0# set nssa default-lsa metric-type 2

{master}[edit protocols ospf area 0.0.0.2]
ibm@J08E-re0# set nssa default-lsa default-metric 20

{master}[edit protocols ospf area 0.0.0.2]
ibm@J08E-re0# set nssa default-lsa type-7

{master}[edit protocols ospf area 0.0.0.2]
ibm@J08E-re0# set nterface ge-1/0/3.0 authentication md5 1 key key-3
```

```
{master}[edit protocols ospf area 0.0.0.2]
ibm@J08E-re0# up

{master}[edit protocols ospf]
ibm@J08E-re0# set reference-bandwidth 100g

{master}[edit protocols ospf]
ibm@J08E-re0# exit

{master}[edit]
ibm@J08E-re0# show protocols ospf
reference-bandwidth 100g;
area 0.0.0.0 {
    interface ge-1/0/1.0 {
        metric 10;
        hello-interval 1;
    }
}
area 0.0.0.1 {
    stub default-metric 10 no-summaries;
    interface ge-1/0/2.0 {
        authentication {
            simple-password "$9$GRiqf3nCtORTQEc"; ## SECRET-DATA
        }
    }
}
area 0.0.0.2 {
    nssa {
        default-lsa {
            default-metric 20;
            metric-type 2;
            type-7;
        }
    }
    interface ge-1/0/3.0 {
        authentication {
            md5 1 key "$9$ss4JD.mTz6AiHT3/9B1LxNd2aJGiP5QUD"; ## SECRET-DATA
        }
    }
}

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# run show ospf neighbor
Address          Interface              State     ID               Pri  Dead
172.16.1.3       ge-1/0/1.0             Full      172.16.0.1       128     3
172.16.2.3       ge-1/0/2.0             Full      172.16.0.2       128    33
```

```
172.16.3.3       ge-1/0/3.0             Full      10.1.1.10        128    37

{master}[edit]
ibm@J08E-re0# run show ospf neighbor extensive
Address          Interface              State     ID               Pri  Dead
172.16.1.3       ge-1/0/1.0             Full      172.16.0.1       128     3
  Area 0.0.0.0, opt 0x42, DR 172.16.1.3, BDR 172.16.1.1
  Up 00:12:00, adjacent 00:12:00
  Topology default (ID 0) -> Bidirectional
172.16.2.3       ge-1/0/2.0             Full      172.16.0.2       128    38
  Area 0.0.0.1, opt 0x40, DR 172.16.2.3, BDR 172.16.2.1
  Up 00:11:59, adjacent 00:11:59
  Topology default (ID 0) -> Bidirectional
172.16.3.3       ge-1/0/3.0             Full      10.1.1.10        128    32
  Area 0.0.0.2, opt 0x40, DR 172.16.3.1, BDR 172.16.3.3
  Up 00:28:43, adjacent 00:28:43
  Topology default (ID 0) -> Bidirectional

{master}[edit]
ibm@J08E-re0# run show ospf interface
Interface           State   Area            DR ID           BDR ID          Nbrs
ge-1/0/1.0          BDR     0.0.0.0         172.16.0.1      10.1.1.1           1
ge-1/0/2.0          BDR     0.0.0.1         172.16.0.2      10.1.1.1           1
ge-1/0/3.0          DR      0.0.0.2         10.1.1.1        10.1.1.10          1

{master}[edit]
ibm@J08E-re0# run show ospf interface extensive
Interface           State   Area            DR ID           BDR ID          Nbrs
ge-1/0/1.0          BDR     0.0.0.0         172.16.0.1      10.1.1.1           1
  Type: LAN, Address: 172.16.1.1, Mask: 255.255.255.0, MTU: 1500, Cost: 10
  DR addr: 172.16.1.3, BDR addr: 172.16.1.1, Priority: 128
  Adj count: 1
  Hello: 1, Dead: 4, ReXmit: 5, Not Stub
  Auth type: None <-----------------------------------------------------no auth
  Protection type: None
  Topology default (ID 0) -> Cost: 0
ge-1/0/2.0          BDR     0.0.0.1         172.16.0.2      10.1.1.1           1
  Type: LAN, Address: 172.16.2.1, Mask: 255.255.255.0, MTU: 1500, Cost: 100
  DR addr: 172.16.2.3, BDR addr: 172.16.2.1, Priority: 128
  Adj count: 1
  Hello: 10, Dead: 40, ReXmit: 5, Stub <---------------------- area type stub
  Auth type: Password <--------------------------------------------simple auth
  Protection type: None
  Topology default (ID 0) -> Cost: 0
ge-1/0/3.0          DR      0.0.0.2         10.1.1.1        10.1.1.10          1
  Type: LAN, Address: 172.16.3.1, Mask: 255.255.255.0, MTU: 1500, Cost: 100
  DR addr: 172.16.3.1, BDR addr: 172.16.3.3, Priority: 128
  Adj count: 1
  Hello: 10, Dead: 40, ReXmit: 5, Stub NSSA <---------------------- area type NSSA
  Auth type: MD5, Active key ID: 1, Start time: 1970 Jan  1 00:00:00 UTC<-md5 auth
  Protection type: None
  Topology default (ID 0) -> Cost: 0

{master}[edit]
ibm@J08E-re0# run show ospf database
```

```
    OSPF database, Area 0.0.0.0
 Type       ID               Adv Rtr           Seq      Age  Opt  Cksum  Len
Router  *10.1.1.1         10.1.1.1         0x80000008   499  0x22 0x1e68  36
Router   10.1.1.10        10.1.1.10        0x80000003  1917  0x22 0x9fd9  36
Router   172.16.0.1       172.16.0.1       0x80000002   913  0x22 0x444b  60
Router   172.16.1.3       172.16.1.3       0x80000006   913  0x22 0x97b0  60
Network  172.16.1.3       172.16.0.1       0x80000002   913  0x22 0x2fc0  32
Summary *172.16.2.0       10.1.1.1         0x80000009   498  0x22 0x5aaa  28
Summary  172.16.2.0       10.1.1.10        0x80000003  2011  0x22 0x30d1  28
Summary *172.16.3.0       10.1.1.1         0x80000005   498  0x22 0x57b0  28
Summary  172.16.3.0       10.1.1.10        0x80000004  1912  0x22 0x23dc  28
ASBRSum *10.1.1.10        10.1.1.1         0x80000006   498  0x22 0xefbf  28

    OSPF database, Area 0.0.0.1
 Type       ID               Adv Rtr           Seq      Age  Opt  Cksum  Len
Router  *10.1.1.1         10.1.1.1         0x80000009   499  0x20 0xa488  36
Router   10.1.1.10        10.1.1.10        0x80000005  1644  0x20 0x24fa  36
Router   172.16.0.2       172.16.0.2       0x80000002   913  0x20 0xa629  36
Router   172.16.2.3       172.16.2.3       0x80000005   913  0x20 0xec9d  36
Network  172.16.2.3       172.16.0.2       0x80000002   913  0x20 0x46a8  32
Summary *0.0.0.0          10.1.1.1         0x80000004   499  0x20 0x968e  28
Summary  172.16.3.0       10.1.1.10        0x80000004  1912  0x20 0x41c0  28

    OSPF database, Area 0.0.0.2
 Type       ID               Adv Rtr           Seq      Age  Opt  Cksum  Len
Router  *10.1.1.1         10.1.1.1         0x80000006   499  0x20 0xb27b  36
Router   10.1.1.10        10.1.1.10        0x80000007   914  0x20 0x39e0  36
Network *172.16.3.1       10.1.1.1         0x80000002   811  0x20 0x1139  32
Summary *172.16.0.0       10.1.1.1         0x80000002   252  0x20 0x1555  28
Summary *172.16.0.1       10.1.1.1         0x80000002    80  0x20 0xb5e   28
Summary *172.16.1.0       10.1.1.1         0x80000006   912  0x20 0x263   28
Summary *172.16.2.0       10.1.1.1         0x80000009   498  0x20 0x788e  28
NSSA    *0.0.0.0          10.1.1.1         0x80000004   499  0x20 0x5638  36
NSSA     192.168.20.0     10.1.1.10        0x80000002   537  0x28 0xf55b  36
NSSA     192.168.21.0     10.1.1.10        0x80000001  1644  0x28 0xec64  36
NSSA     192.168.22.0     10.1.1.10        0x80000001  1644  0x28 0xe16e  36
NSSA     192.168.23.0     10.1.1.10        0x80000001  1644  0x28 0xd678  36
    OSPF AS SCOPE link state database
 Type       ID               Adv Rtr           Seq      Age  Opt  Cksum  Len
Extern  *192.168.20.0     10.1.1.1         0x80000002   498  0x22 0xa2bf  36
Extern  *192.168.21.0     10.1.1.1         0x80000002   498  0x22 0x97c9  36
Extern  *192.168.22.0     10.1.1.1         0x80000002   498  0x22 0x8cd3  36
Extern  *192.168.23.0     10.1.1.1         0x80000002   498  0x22 0x81dd  36

{master}[edit]
ibm@J08E-re0# run show route protocol ospf

inet.0: 24 destinations, 26 routes (24 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.0.0/24      *[OSPF/10] 00:15:37, metric 10
                    > to 172.16.1.3 via ge-1/0/1.0
172.16.0.1/32      *[OSPF/10] 00:15:37, metric 10
                    > to 172.16.1.3 via ge-1/0/1.0
```

```
192.168.20.0/24    *[OSPF/150] 00:27:47, metric 0, tag 2
                    > to 172.16.3.3 via ge-1/0/3.0
192.168.21.0/24    *[OSPF/150] 00:27:47, metric 0, tag 2
                    > to 172.16.3.3 via ge-1/0/3.0
192.168.22.0/24    *[OSPF/150] 00:27:47, metric 0, tag 2
                    > to 172.16.3.3 via ge-1/0/3.0
192.168.23.0/24    *[OSPF/150] 00:27:47, metric 0, tag 2
                    > to 172.16.3.3 via ge-1/0/3.0
224.0.0.5/32       *[OSPF/10] 00:37:29, metric 1
                      MultiRecv

inet6.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

{master}[edit]
ibm@J08E-re0# run show route 192.168.20.0/24 extensive

inet.0: 24 destinations, 26 routes (24 active, 0 holddown, 0 hidden)
192.168.20.0/24 (1 entry, 1 announced)
TSI:
KRT in-kernel 192.168.20.0/24 -> {172.16.3.3}
        *OSPF   Preference: 150
                Next hop type: Router, Next hop index: 1346
                Next-hop reference count: 8
                Next hop: 172.16.3.3 via ge-1/0/3.0, selected
                State: <Active Int Ext>
                Age: 28:47 Metric: 0 Tag: 2
                Task: OSPF
                Announcement bits (1): 0-KRT
                AS path: I

{master}[edit]
ibm@J08E-re0# run show route 172.16.0.1/32 extensive

inet.0: 24 destinations, 26 routes (24 active, 0 holddown, 0 hidden)
172.16.0.1/32 (1 entry, 1 announced)
TSI:
KRT in-kernel 172.16.0.1/32 -> {172.16.1.3}
        *OSPF   Preference: 10
                Next hop type: Router, Next hop index: 1345
                Next-hop reference count: 4
                Next hop: 172.16.1.3 via ge-1/0/1.0, selected
                State: <Active Int>
                Age: 17:04 Metric: 10
                Area: 0.0.0.0
                Task: OSPF
                Announcement bits (1): 0-KRT
                AS path: I

{master}[edit]
ibm@J08E-re0#
```

10.3.3 BGP

BGP is an Exterior Gateway Protocol (EGP) that is used to exchange routing information among routers in different autonomous systems. BGP routing information includes the complete route to each destination. BGP uses the routing information to maintain a database of network reachability information, which it exchanges with other BGP systems. BGP uses the network reachability information to construct a graph of AS connectivity. This information allows BGP to remove routing loops and enforce policy decisions at the AS level.

Multiprotocol BGP (MBGP) extensions enable BGP to support IP version 6 (IPv6). MBGP defines the MP_REACH_NLRI and MP_UNREACH_NLRI attributes, which are used to carry IPv6 reachability information. Network layer reachability information (NLRI) update messages carry IPv6 address prefixes of feasible routes.

BGP allows for policy-based routing. You can use routing policies to choose among multiple paths to a destination and to control the redistribution of routing information.

BGP uses TCP as its transport protocol, using port 179 for establishing connections. Running over a reliable transport protocol eliminates the need for BGP to implement update fragmentation, retransmission, acknowledgment, and sequencing.

The Junos routing protocol software supports BGP version 4. This version of BGP adds support for Classless Interdomain Routing (CIDR), which eliminates the concept of network classes. You do not need to assume which bits of an address represent the network by looking at the first octet. Instead, with CIDR, you can explicitly specify the number of bits in the network address, providing a means to decrease the size of the routing tables. BGP version 4 also supports the aggregation of routes, including the aggregation of AS paths.

BGP routers and routing

This section includes the following topics:

- Autonomous systems
- AS paths and attributes
- External and internal BGP
- Overview of BGP routes
- Selecting the active route

Autonomous systems

An AS is a set of routers that are under a single technical administration. They normally use a single IGP and a common set of metrics to propagate routing information within the set of routers. To other autonomous systems, an AS has a single, coherent interior routing plan and presents a consistent picture of which destinations are reachable through it.

AS paths and attributes

The routing information that BGP systems exchange includes the complete route to each destination and additional information about the route. The route to each destination is called the *AS path*, and the additional route information is included in path attributes. BGP uses the AS path and the path attributes to completely determine the network topology. When BGP understands the topology, it can detect and eliminate routing loops and select among groups of routes to enforce administrative preferences and routing policy decisions.

External and internal BGP

BGP supports two types of exchanges of routing information: exchanges between different autonomous systems and exchanges within a single AS. When used between autonomous systems, BGP is called *external BGP* (EBGP), and BGP sessions perform *inter-AS routing*.

When used within an AS, BGP is called *internal BGP* (IBGP), and BGP sessions perform *intra-AS routing*. Figure 10-3 illustrates autonomous systems, IBGP, and EBGP.

Figure 10-3 BGP AS, IBGP, and EBGP

A BGP system shares network reachability information with adjacent BGP systems, which are referred to as *neighbors* or *peers*.

BGP systems are arranged into *groups*. In an IBGP group, all peers in the group, called *internal peers*, are in the same AS. Internal peers can be anywhere in the local AS and do not have to be directly connected to each other. Internal groups use routes from an IGP to resolve forwarding addresses. They also propagate external routes among all other internal routers running IBGP. They compute the next hop by taking the BGP next hop received with the route and resolve it by using information from one of the IGPs.

In an EBGP group, the peers in the group, called *external peers*, are in different autonomous systems and normally share a subnet. In an external group, the next hop is computed with respect to the interface that is shared between the external peer and the local router.

Overview of BGP routes

A BGP route consists of a destination, described as an IP address prefix, and information that describes the path to the destination. This information includes the AS path, which is a list of numbers of the autonomous systems that a route passes through to reach the local router. The first number in the path is that of the last AS in the path, the AS closest to the local router. The last number in the path is the AS farthest from the local router, which is generally the origin of the path. The information also includes the path attributes, which contain additional information about the AS path that is used in the routing policy.

BGP peers advertise routes to each other in update messages. BGP stores its routes in the Junos software routing table. The routing table stores the following information about BGP routes:

- Routing information learned from update messages received from peers
- Local routing information that the BGP system selects by applying local policies to routes received in update messages

- Information that the BGP system selects to advertise to its BGP peers in the update messages it sends

For each prefix in the routing table, the routing protocol process selects a single best path, called the *active path*.

Selecting the active route

For each prefix in the routing table, the routing protocol process selects a single best path, called the *active route*. Determining the active route entails the following algorithm:

1. Select the path with the lowest preference value (routing protocol process preference). Routes that are not eligible to be used for forwarding (for example, they were rejected by routing policy or a next hop is inaccessible) have a preference of -1 and are never selected.
2. For BGP, prefer the path with higher local preference. For non-BGP paths, choose the path with the lowest preference value.
3. If the path includes an AS path:
   a. Prefer the route with a shorter AS path. Confederation sequences are considered to have a path length of 0, and AS and confederation sets are considered to have a path length of 1.
   b. Prefer the route with the lower origin code. Routes learned from an IGP have a lower origin code than routes learned from an EGP. Both have lower origin codes than incomplete routes (routes whose origin is unknown).
   c. Depending on whether the path selection behavior of a nondeterministic routing table is configured, two cases are possible:
      - The path selection behavior of a nondeterministic routing table is not configured (that is, if the **`path-selection cisco-nondeterministic`** statement is not included in the BGP configuration). In this case, for paths with the same neighboring AS numbers at the front of the AS path, prefer the path with the lowest multiple exit discriminator (MED) metric. Confederation AS numbers are not considered when deciding what the neighbor AS number is.
      - To always compare MEDs, regardless of whether the peer autonomous systems of the compared routes are the same, include the **`path-selection always-compare-med`** statement.

      In both cases, confederations are not considered when determining neighboring autonomous systems. Also, in both cases, a missing metric is treated as though an MED were present but with a value of zero.
4. Prefer strictly internal paths, which include IGP routes and locally generated routes (static, direct, local, and so on).
5. Prefer strictly external (EBGP) paths over external paths learned through interior sessions (IBGP).
6. For BGP, prefer the path whose next hop is resolved through the IGP route with the lowest metric.
7. For BGP, if both paths are external, prefer the currently active path to minimize route-flapping. This rule is not used in the following cases:
   – **`path-selection external-router-id`** is configured.
   – Both peers have the same router ID.
   – Either peer is a confederation peer.
   – Neither path is the current active path.

8. For BGP, prefer the path from the peer with the lowest router ID. For any path with an originator ID attribute, substitute the originator ID for the router ID during router ID comparison.
9. For BGP, prefer the path with the shortest cluster list length. The length is 0 for no list.
10. For BGP, prefer the path from the peer with the lowest peer IP address.

BGP standards

The Junos software supports BGP version 4 and several extensions to the protocol, which are defined in the following documents:

- RFC 1772, Application of the Border Gateway Protocol in the Internet
- RFC 1965, Autonomous System Confederations for BGP
- RFC 1966, BGP Route Reflection: An Alternative to Full-Mesh IBGP
- RFC 1997, BGP Communities Attribute
- RFC 2270, Using a Dedicated AS for Sites Homed to a Single Provider
- RFC 2283, Multiprotocol Extensions for BGP-4
- RFC 2385, Protection of BGP Sessions via the TCP MD5 Signature Option
- RFC 2439, BGP Route Flap Damping
- RFC 2545, Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing
- RFC 2796, BGP Route Reflection
- RFC 2858, Multiprotocol Extensions for BGP-4
- RFC 2918, Route Refresh Capability for BGP-4
- RFC 3065, Autonomous System Confederations for BGP
- RFC 3107, Carrying Label Information in BGP-4
- RFC 3392, Capabilities Advertisement with BGP-4
- RFC 4271, A Border Gateway Protocol 4 (BGP-4)
- RFC 4724, Graceful Restart Mechanism for BGP
- RFC 4781, Graceful Restart Mechanism for BGP with MPLS
- RFC 4893, BGP Support for Four-octet AS Number Space
- Internet draft `draft-ietf-ppvpn-rfc2547bis-00.txt`, BGP/MPLS VPNs (expires January 2002)
- Internet draft `draft-kato-bgp-ipv6-link-local-00.txt`, BGP4+ Peering Using IPv6 Link-local Address (expires April 2002)
- Internet draft `draft-ietf-ngtrans-bgp-tunnel-04.txt`, Connecting IPv6 Islands across IPv4 Clouds with BGP (only MP-BGP over IPv4 Approach) (expires July 2002)
- Internet draft `draft-ietf-idr-flow-spec-00.txt`, Dissemination of Flow Specification Rules

To access Internet RFCs and drafts, go to the IETF website at:

http://www.ietf.org

BGP configuration

This section provides information about the configuration options for BGP on IBM j-type Ethernet switches and routers.

Minimum BGP configuration

For BGP to run on the routing device, you must define the local AS number and configure at least one group. You must also include information about at least one peer in the group (the peer's IP address and AS number).

BGP options: Most BGP options can be specified at different configuration hierarchy levels. The most specific hierarchy has precedence over the least specific ones.

Specifying the AS number of the local routing device

Each routing device running BGP must be configured with its AS number. This number is included in the local AS number field in BGP open messages, which are sent between BGP peers to establish a connection.

To specify an AS number, use the following command:

```
{master}[edit]
ibm@J08E-re0# set routing-options autonomous-system as_number
```

Configuring BGP groups and peers

A BGP system must know which routing devices are its peers (neighbors). You define the peer relationships explicitly by configuring the neighboring routers that are the peers of the local BGP system. After peer relationships have been established, the BGP peers exchange update messages to advertise network reachability information.

You arrange BGP routing devices into groups of peers. Peer groups must have different group types, AS numbers, or route reflector cluster identifiers. Each group must contain at least one peer.

To define a BGP group that recognizes only the specified BGP systems as peers, statically configure all the system's peers by including one or more neighbor statements. The peers on at least one side of each BGP connection must be configured statically. The peer neighbor's address can be either an IPv6 or IPv4 address.

To create an external BGP group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group type external
```

To create an internal BGP group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group type internal
```

To add neighbors to the newly created BGP groups, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group neighbor neighbor
```

To specify the peer-as number of a BGP neighbor, you can use the **peer-as** command at a group level or at a neighbor level. To configure at the group level, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group peer-as peer-as
```

To configure at the neighbor level, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group neighbor neighbor peer-as peer-as
```

> **BGP options:** Most BGP options can be specified at different configuration hierarchy levels. The most specific hierarchy has precedence over the least specific ones. As a good practice, group similar BGP neighbors in a BGP group and specify the options at a group level.

In this case, the peer-as specified at neighbor level takes precedence over the peer-as specified at group level.

Configuring the delay before BGP peers mark the routing device as down

The hold time is the maximum number of seconds that are allowed to elapse between successive keepalive or update messages that BGP receives from a peer. When establishing a BGP connection with the local routing device, a peer sends an open message, which contains a hold-time value. BGP on the local routing device uses the smaller value (either the local hold-time value or the peer's hold-time value) received in the open message as the hold time for the BGP connection between the two peers.

You can configure hold time at the BGP level, group level, and neighbor level by using the following commands:

- For the BGP level, affecting all BGP sessions, use the following command:

  ```
  {master}[edit]
  ibm@J08E-re0# set protocols bgp hold-time time
  ```

- For the BGP group level, affecting all BGP sessions from this BGP group, use the following command:

  ```
  {master}[edit]
  ibm@J08E-re0# set protocols bgp group group hold-time time
  ```

- For the BGP neighbor level, affecting only this specific BGP neighbor, use the following command:

  ```
  {master}[edit]
  ibm@J08E-re0# set protocols bgp group group neighbor neighbor hold-time time
  ```

The default `hold-time` value is 90 seconds. The range is 20–65,535 seconds. The hold time is three times the interval at which keepalive messages are sent.

As a good practice, group similar BGP neighbors in a BGP group and specify the options at a group level.

Configuring MTU discovery for BGP sessions

You can configure TCP path maximum transmission unit (MTU) discovery. MTU discovery improves convergence times for IBGP sessions. BGP unconditionally disables TCP path MTU discovery, resulting in a 512 byte maximum segment size (MSS) on TCP sessions that are not directly connected. With this feature, you can enable TCP path MTU discovery on BGP sessions.

To configure MTU discovery, include the **mtu-discovery** statement. You can configure MTU discovery at the BGP level, group level, and neighbor level by using the following commands:

- For the BGP level, affecting all BGP sessions, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp mtu-discovery
```

- For the BGP group level, affecting all BGP sessions from this BGP group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group mtu-discovery
```

- For the BGP neighbor level, affecting only this specific BGP neighbor, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group neighbor neighbor mtu-discovery
```

As a good practice, group similar BGP neighbors in a BGP group and specify the options at a group level.

Configuring authentication for BGP

All BGP protocol exchanges can be authenticated to guarantee that only trusted routing devices participate in routing of the AS. By default, authentication is disabled.

You can configure MD5 authentication. The MD5 algorithm creates an encoded checksum that is included in the transmitted packet. The receiving routing device uses an authentication key (password) to verify the MD5 checksum of the packet.

To configure an MD5 authentication key, include the **authentication-key** statement. You can configure MD5 authentication at the BGP level, group level, and neighbor level by using the following commands:

- For the BGP level, affecting all BGP sessions, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp authentication-key key
```

- For the BGP group level, affecting all BGP sessions from this BGP group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group authentication-key key
```

- For the BGP neighbor level, affecting only this specific BGP neighbor, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group neighbor neighbor
authentication-key key
```

The key (password) can be up to 126 characters long. Characters can include any ASCII strings. If you include spaces, enclose all characters in quotation marks (“ ”).

As a good practice, group similar BGP neighbors in a BGP group, and specify the options at a group level.

Configuring a local endpoint address for BGP sessions

You can specify the address of the local end of a BGP session. In general, you specify the address to explicitly configure the IP address of the system from point of view of the BGP. This IP address can be either an IPv6 or IPv4 address. Typically, an IP address is assigned to

a loopback interface, and that IP address is configured here. This address is used to accept incoming connections to the peer and to establish connections to the remote peer. To assign a local address, include the **local-address** statement.

You can configure local address at the BGP level, group level, and neighbor level by using the following commands:

- For the BGP level, affecting all BGP sessions, use the following command:

  ```
  {master}[edit]
  ibm@J08E-re0# set protocols bgp local-address address
  ```

- For the BGP group level, affecting all BGP sessions from this BGP group, use the following command:

  ```
  {master}[edit]
  ibm@J08E-re0# set protocols bgp group group local-address address
  ```

- For the BGP neighbor level, affecting only this specific BGP neighbor, use the following command:

  ```
  {master}[edit]
  ibm@J08E-re0# set protocols bgp group group neighbor neighbor local-address
  address
  ```

BGP session: A BGP session can still be established when only one of the paired routers has a local address configured.

As a good practice, group similar BGP neighbors in a BGP group, and specify the options at a group level.

Configuring the MED in BGP updates

The BGP multiple exit discriminator (MED or MULTI_EXIT_DISC) is an optional path attribute that can be included in BGP update messages. This attribute is used on external BGP links (that is, on inter-AS links) to select among multiple exit points to a neighboring AS. The MED attribute has a value that is referred to as a *metric*. If all other factors in determining an exit point are equal, the exit point with the lowest metric is preferred.

If an MED is received over an external BGP link, it is propagated over internal links to other BGP systems within the AS.

BGP update messages include an MED metric if the route was learned from BGP and already had a MED metric associated with it. Alternatively, you can directly configure the MED metric.

An MED metric is advertised with a route according to the following general rules:

- A more specific metric overrides a less specific metric. That is, a group-specific metric overrides a global BGP metric, and a peer-specific metric overrides a global BGP or group-specific metric.
- A metric defined with routing policy overrides a metric defined with the metric-out statement.
- If any metric is defined, it overrides a metric received in a route.
- If the received route does not have an associated MED metric, and if you do not explicitly configure a metric value, no metric is advertised. When you do not explicitly configure a metric value, the MED is equivalent to 0 when advertising an active route.

To directly configure an MED metric to advertise in BGP update messages, include the **metric-out** statement. You can configure MED metric at the BGP level, group level, and neighbor level by using the following commands:

- For the BGP level, affecting all BGP sessions, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp metric-out metric
```

- For the BGP group level, affecting all BGP sessions from this BGP group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group metric-out metric
```

- For the BGP neighbor level, affecting only this specific BGP neighbor, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group neighbor neighbor metric-out metric
```

You can also use other ways to modify the MED metric of advertised BGP routes. For detailed information, see appropriate document in 10.7, “More information” on page 330.

Configuring EBGP multihop sessions

If an EBGP peer is more than one hop away from the local router, specify the next hop to the peer so that the two systems can establish a BGP session. This type of session is called a *multihop BGP session*.

To configure a multihop session, include the **multihop** statement. You can configure multihop at the BGP level, group level, and neighbor level by using the following commands:

- For the BGP level, affecting all BGP sessions, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp multihop [TTL]
```

- For the BGP group level, affecting all BGP sessions from this BGP group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group multihop [TTL]
```

- For the BGP neighbor level, affecting only this specific BGP neighbor, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group neighbor neighbor multihop [TTL]
```

As a good practice, configure BGP multihop at the neighbor level.

To configure the maximum time-to-live (TTL) value for the TTL in the IP header of BGP packets, specify the TTL value. If you do not specify a TTL value, the default maximum TTL value of the system is used.

Configuring the local preference value for BGP routes

IBGP sessions use a metric called the *local preference*, which is carried in IBGP update packets in the path attribute LOCAL_PREF. This metric indicates the degree of preference for an external route. The route with the highest local preference value is preferred.

The LOCAL_PREF path attribute is always advertised to internal BGP peers and to neighboring confederations. It is never advertised to external BGP peers. The default behavior is to not modify the LOCAL_PREF path attribute if it is present.

By default, if a received route contains a LOCAL_PREF path attribute value, the value is not modified. If a BGP route is received without a LOCAL_PREF attribute, the route is handled locally (that is, stored in the routing table and advertised by BGP) as though it were received with a LOCAL_PREF value of 100. A non-BGP route that is advertised by BGP is advertised with a LOCAL_PREF value of 100 by default.

To configure a local preference metric, include the **local-preference** statement. You can configure local-preference at the BGP level, group level, and neighbor level by using the following commands:

- For the BGP level, affecting all BGP sessions, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp local-preference preference_value
```

- For the BGP group level, affecting all BGP sessions from this BGP group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group local-preference preference_value
```

- For the BGP neighbor level, affecting only this specific BGP neighbor, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group neighbor neighbor local-preference
preference_value
```

As a good practice, group similar BGP neighbors in a BGP group, and specify the options at a group level.

Configuring the default preference value for BGP routes

When the Junos software determines the preference of a route to become the active route, it selects the route with the lowest preference as the active route. Then it installs this route into the forwarding table. By default, the routing software assigns a preference of 170 to routes that originated from BGP. Of all the routing protocols, BGP has the highest default preference value. The routes learned by BGP are the least likely to become the active route.

To modify the default BGP preference value, include the **preference** statement. You can configure the BGP routes preference at the BGP level, group level, and neighbor level by using the following commands:

- For the BGP level, affecting all BGP sessions, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp preference preference_value
```

- For the BGP group level, affecting all BGP sessions from this BGP group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group preference preference_value
```

- For the BGP neighbor level, affecting only this specific BGP neighbor, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols bgp group group neighbor neighbor preference
preference_value
```

As a good practice, group similar BGP neighbors in a BGP group, and specify the options at a group level.

Configuring BGP route reflection

In standard IBGP implementations, all BGP systems within the AS are fully meshed so that any external routing information is redistributed among all routing devices within the AS. This type of implementation can present scaling issues when an AS has many internal BGP systems because of the amount of identical information that BGP systems must share with each other. Route reflection provides one means of decreasing BGP control traffic and minimizing the number of update messages sent within the AS.

In route reflection, BGP systems are arranged in clusters. Each cluster consists of at least one system that acts as a route reflector, along with any number of client peers.

BGP peers outside the cluster are called *nonclient peers*. The route reflector reflects (re-distributes) routing information to each client peer (intracluster reflection) and to all nonclient peers (intercluster reflection). Because the route reflector redistributes routes within the cluster, the BGP systems within the cluster do not have to be fully meshed.

When the route reflector receives a route, it selects the best path. Then, if the route came from a nonclient peer, the route reflector sends the route to all client peers within the cluster. If the route came from a client peer, the route reflector sends it to all nonclient peers and to all client peers except the originator. In this process, none of the client peers send routes to other client peers.

To configure route reflection, you specify a cluster identifier only on the BGP systems that are to be the route reflectors. These systems then determine, from the network reachability information that they receive, which BGP systems are part of its cluster and are client peers, and which BGP systems are outside the cluster and are nonclient peers.

To configure the route reflector, include the **`cluster`** identifier statement in the configuration. You can configure the BGP routes preference at the BGP level, group level, and neighbor level by using the following commands:

- For the BGP level, affecting all BGP sessions, use the following command:

  ```
  {master}[edit]
  ibm@J08E-re0# set protocols bgp cluster cluster_identifier
  ```

- For the BGP group level, affecting all BGP sessions from this BGP group, use the following command:

  ```
  {master}[edit]
  ibm@J08E-re0# set protocols bgp group group cluster cluster_identifier
  ```

- For the BGP neighbor level, affecting only this specific BGP neighbor, use the following command:

  ```
  {master}[edit]
  ibm@J08E-re0# set protocols bgp group group neighbor neighbor cluster
  cluster_identifier
  ```

As a good practice, group similar BGP neighbors in a BGP group, and specify the options at a group level.

BGP route reflector: You must only configure the BGP route reflector (BGP-RR) with this command. The BGP-RR clients have a normal configuration.

Assigning a BGP identifier

Each routing device running BGP must have a BGP identifier. This identifier is included in the BGP identifier field of open messages, which are sent between two BGP peers when establishing a BGP session.

Explicitly assigning a BGP identifier is optional. If you do not assign one, the IP address of the first interface encountered in the routing device is used.

To assign a BGP identifier explicitly, use the following command:

```
{master}[edit]
ibm@J08E-re0# set routing-options router-id router-id
```

Defining an AS confederation and its members

To enable the local system to participate as a member of an AS confederation, you must define the AS confederation identifier. You must also specify the AS numbers that are members of the confederation. you can define the identifier and specify the numbers, use the following command:

```
{master}[edit]
ibm@J08E-re0# set routing-options confederation confederation_as_number members
as_members
```

Verifying BGP operations

To verify BGP operation, use any of the following commands:

```
{master}[edit]
ibm@J08E-re0> show bgp summary

{master}[edit]
ibm@J08E-re0> show bgp neighbor

{master}[edit]
ibm@J08E-re0> show route protocol bgp [terse | extensive]
```

Many other options are available for configuring and monitoring BGP on IBM j-type series switches and routers. For more details, see the appropriate documents in 10.7, “More information” on page 330.

Example 10-3 on page 313 shows the configuration of three BGP groups with the following characteristics:

- BGP group **ibgp**
  - Internal BGP peering
  - AS of 65000
  - Loopback address for peering (10.0.0.1 is our loopback)
  - IBGP neighbors 10.0.0.2 and 10.0.0.3
  - A route reflector with a cluster-id of 1
- BGP group **ebgp1**
  - External BGP peering
  - EBGP neighbor of 172.16.1.3
  - External AS of 65001
- BGP group **ebgp4**
  - External BGP peering
  - EBGP neighbor of 172.16.4.3
  - External AS of 65004

– Authentication of this BGP session with MD5 key "key-1"
– A hold-time of 20 seconds
– Preference of 169 to prefer the routes from this EBGP group (default is 170)

Example 10-3 BGP configuration

```
[edit]
ibm@J02M# edit protocols bgp

[edit protocols bgp]
ibm@J02M# set group ibgp type internal

[edit protocols bgp]
ibm@J02M# set group ibgp local-address 10.0.0.1

[edit protocols bgp]
ibm@J02M# set group ibgp cluster 1

[edit protocols bgp]
ibm@J02M# set group ibgp peer-as 65000

[edit protocols bgp]
ibm@J02M# set group ibgp neighbor 10.0.0.2

[edit protocols bgp]
ibm@J02M# set group ibgp neighbor 10.0.0.3

[edit protocols bgp]
ibm@J02M# set group ebgp1 type external

[edit protocols bgp]
ibm@J02M# set group ebgp1 peer-as 65001

[edit protocols bgp]
ibm@J02M# set group ebgp1 neighbor 172.16.1.3

[edit protocols bgp]
ibm@J02M# set group ebgp4 type external

[edit protocols bgp]
ibm@J02M# set group ebgp4 preference 169

[edit protocols bgp]
ibm@J02M# set group ebgp4 peer-as 65004

[edit protocols bgp]
ibm@J02M# set group ebgp4 neighbor 172.16.4.3 hold-time 20

[edit protocols bgp]
ibm@J02M# set group ebgp4 neighbor 172.16.4.3 authentication-key key-1

[edit protocols bgp]
ibm@J02M# top set routing-options autonomous-system 65000 <---- set our own AS#

[edit protocols bgp]
ibm@J02M# show
```

```
group ibgp {
    type internal;
    local-address 10.0.0.1;
    cluster 0.0.0.1;
    peer-as 65000;
    neighbor 10.0.0.2;
    neighbor 10.0.0.3;
}
group ebgp1 {
    type external;
    peer-as 65001;
    neighbor 172.16.1.3;
}
group ebgp4 {
    type external;
    preference 169;
    peer-as 65004;
    neighbor 172.16.4.3 {
        hold-time 20;
        authentication-key "$9$.5z6pu1RSe9Cev"; ## SECRET-DATA
    }
}

[edit protocols bgp]
ibm@J02M# commit
commit complete

[edit protocols bgp]
ibm@J02M# run show bgp summary
Groups: 3 Peers: 4 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
inet.0                 8          4          0          0          0          0
Peer                     AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn
State|#Active/Received/Accepted/Damped...
10.0.0.2              65000         83         84       0       0       36:19
0/0/0/0              0/0/0/0
10.0.0.3              65000         82         83       0       0       36:15
0/0/0/0              0/0/0/0
172.16.1.3            65001         59         60       0       0       25:48
4/4/4/0              0/0/0/0
172.16.4.3            65004        291        291       0       0       25:48
0/4/4/0              0/0/0/0

ibm@J02M# run show bgp neighbor 172.16.4.3
Peer: 172.16.4.3+65184 AS 65004 Local: 172.16.4.1+179 AS 65000
  Type: External    State: Established    Flags: <Sync>
  Last State: OpenConfirm   Last Event: RecvKeepAlive
  Last Error: None
  Options: <Preference HoldTime AuthKey Localpref PeerAS Refresh>
  Authentication key is configured <---------------- Authentication configured
  Holdtime: 20 Preference: 169 <--------------- Holdtime and preference values
  Number of flaps: 0
  Peer ID: 172.16.4.3       Local ID: 10.0.0.1         Active Holdtime: 20
  Keepalive Interval: 6          Peer index: 0
  BFD: disabled, down
```

```
  Local Interface: ge-1/0/0.4
  NLRI for restart configured on peer: inet-unicast
  NLRI advertised by peer: inet-unicast
  NLRI for this session: inet-unicast
  Peer supports Refresh capability (2)
  Restart time configured on the peer: 120
  Stale routes from peer are kept for: 300
  Restart time requested by this peer: 120
  NLRI that peer supports restart for: inet-unicast
  NLRI that restart is negotiated for: inet-unicast
  NLRI of received end-of-rib markers: inet-unicast
  NLRI of all end-of-rib markers sent: inet-unicast
  Peer supports 4 byte AS extension (peer-as 65004)
  Peer does not support Addpath
  Table inet.0 Bit: 10002
    RIB State: BGP restart is complete
    Send state: in sync
    Active prefixes:              4
    Received prefixes:            4
    Accepted prefixes:            4
    Suppressed due to damping:    0
    Advertised prefixes:          0
  Last traffic (seconds): Received 1    Sent 5    Checked 3
  Input messages:  Total 442Updates 3Refreshes 0Octets 8522
  Output messages: Total 443Updates 2Refreshes 1Octets 8548
  Output Queue[0]: 0

[edit protocols bgp]
ibm@J02M# run show bgp neighbor 172.16.1.3
Peer: 172.16.1.3+179 AS 65001  Local: 172.16.1.1+58640 AS 65000
  Type: External    State: Established    Flags: <Sync>
  Last State: OpenConfirm   Last Event: RecvKeepAlive
  Last Error: None
  Options: <Preference PeerAS Refresh>
  Holdtime: 90 Preference: 170 <--------- Default holdtime and preference values
  Number of flaps: 0
  Peer ID: 172.16.1.3       Local ID: 10.0.0.1         Active Holdtime: 90
  Keepalive Interval: 30         Peer index: 0
  BFD: disabled, down
  Local Interface: ge-1/0/0.1
  NLRI for restart configured on peer: inet-unicast
  NLRI advertised by peer: inet-unicast
  NLRI for this session: inet-unicast
  Peer supports Refresh capability (2)
  Restart time configured on the peer: 120
  Stale routes from peer are kept for: 300
  Restart time requested by this peer: 120
  NLRI that peer supports restart for: inet-unicast
  NLRI that restart is negotiated for: inet-unicast
  NLRI of received end-of-rib markers: inet-unicast
  NLRI of all end-of-rib markers sent: inet-unicast
  Peer supports 4 byte AS extension (peer-as 65001)
  Peer does not support Addpath
  Table inet.0 Bit: 10001
    RIB State: BGP restart is complete
```

```
    Send state: in sync
    Active prefixes:              0
    Received prefixes:            4
    Accepted prefixes:            4
    Suppressed due to damping:    0
    Advertised prefixes:          4
  Last traffic (seconds): Received 19   Sent 17   Checked 68
  Input messages:  Total 93Updates 3Refreshes 0 Octets 1851
  Output messages: Total 95Updates 1Refreshes 1 Octets 1916
  Output Queue[0]: 0

[edit protocols bgp]
ibm@J02M# run show route protocol bgp terse

inet.0: 19 destinations, 23 routes (18 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 192.168.1.0/24     B 169        100            >172.16.4.3      65004 I
                     B 170        100            >172.16.1.3      65001 I
* 192.168.2.0/24     B 169        100            >172.16.4.3      65004 I
                     B 170        100            >172.16.1.3      65001 I
* 192.168.3.0/24     B 169        100            >172.16.4.3      65004 I
                     B 170        100            >172.16.1.3      65001 I
* 192.168.4.0/24     B 169        100            >172.16.4.3      65004 I
                     B 170        100            >172.16.1.3      65001 I

[edit protocols bgp]
ibm@J02M# run show route extensive 192.168.1.0/24

inet.0: 19 destinations, 23 routes (18 active, 0 holddown, 1 hidden)
192.168.1.0/24 (2 entries, 1 announced)
TSI:
KRT in-kernel 192.168.1.0/24 -> {172.16.4.3}
Page 0 idx 0 Type 1 val 8ee1930
    Nexthop: 172.16.4.3
    Localpref: 100
    AS path: [65000] 65004 I
    Communities:
Page 0 idx 1 Type 1 val 8ee1914
    Nexthop: Self
    AS path: [65000] 65004 I
    Communities:
Path 192.168.1.0 from 172.16.4.3 Vector len 4.  Val: 0 1
        *BGP    Preference: 169/-101
                Next hop type: Router, Next hop index: 640
                Next-hop reference count: 12
                Source: 172.16.4.3
                Next hop: 172.16.4.3 via ge-1/0/0.4, selected
                State: <Active Ext>
                Local AS: 65000 Peer AS: 65004
                Age: 41:42
                Task: BGP_65004.172.16.4.3+65184
                Announcement bits (3): 0-KRT 4-BGP RT Background 5-Resolve tree 1
                AS path: 65004 I
```

```
                Accepted
                Localpref: 100
                Router ID: 172.16.4.3
          BGP   Preference: 170/-101
                Next hop type: Router, Next hop index: 639
                Next-hop reference count: 4
                Source: 172.16.1.3
                Next hop: 172.16.1.3 via ge-1/0/0.1, selected
                State: <Ext>
                Inactive reason: Route Preference
                Local AS: 65000 Peer AS: 65001
                Age: 41:42
                Task: BGP_65001.172.16.1.3+179
                AS path: 65001 I
                Accepted
                Localpref: 100
                Router ID: 172.16.1.3

[edit protocols bgp]
ibm@J02M# run show bgp group ibgp
Group Type: Internal    AS: 65000                  Local AS: 65000
  Name: ibgp            Index: 0                   Flags: <>
  Options: <Cluster> <---------------------- The ibgp group is BGP Route Reflector
  Holdtime: 0
  Total peers: 2        Established: 2
  10.0.0.2+57377
  10.0.0.3+49548
  inet.0: 0/0/0/0

[edit protocols bgp]
ibm@J02M#
```

10.4 Static routes

This section provides the configuration options for static routes on IBM j-type Ethernet switches and routers. Static routes are routes that are manually configured and entered into the routing table.

To configure a static route, use the following command:

```
{master}[edit]
ibm@J08E-re0# set routing-options static route route/mask
```

To configure a next-hop for a static route, use the following command:

```
{master}[edit]
ibm@J08E-re0# set routing-options static route route/mask next-hop gateway_ip
```

To always keep the static route in the forwarding table, use the following command:

```
{master}[edit]
ibm@J08E-re0# set routing-options static route route/mask retain
```

To prevent the static route from being re-advertised in a routing protocol, use the following command:

```
{master}[edit]
ibm@J08E-re0# set routing-options static route route/mask no-readvertise
```

To drop the packets to a specific static route and send Internet Control Message Protocol (ICMP) unreachable messages back to the source, use the following command:

```
{master}[edit]
ibm@J08E-re0# set routing-options static route route/mask discard
```

To change the route preference for a static route, use the following command:

```
{master}[edit]
ibm@J08E-re0# set routing-options static route route/mask preference preference
```

To verify the status of the static routes, use the following command:

```
{master}
ibm@J08E-re0> show route protocol static [ terse | extensive ]
```

Example 10-4 shows how to create several static routes and then verify them. First a default gateway is created by using 192.168.1.1 and then a route to 10.0.0.0/8 by using 192.168.1.2. Next a route to 172.16.0.0/12 is created with a preference of 100 that instructs the router to drop the packets and send ICMP unreachable messages back to the source.

Example 10-4 Static routes configuration and verification

```
{master}[edit]
ibm@J08E-re0# set routing-options static route 0.0.0.0/0 next-hop 192.168.1.1

{master}[edit]
ibm@J08E-re0# set routing-options static route 10.0.0.0/8 next-hop 192.168.1.2

{master}[edit]
ibm@J08E-re0# set routing-options static route 172.16.0.0/12 discard

{master}[edit]
ibm@J08E-re0# set routing-options static route 172.16.0.0/12 preference 100

{master}[edit]
ibm@J08E-re0# show routing-options
static {
    route 0.0.0.0/0 next-hop 192.168.1.1;
    route 10.0.0.0/8 next-hop 192.168.1.2;
    route 172.16.0.0/12 {
        discard;
        preference 100;
    }
}

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
```

```
commit complete

{master}[edit]
ibm@J08E-re0# run show route protocol static terse

inet.0: 11 destinations, 13 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination        P Prf   Metric 1   Metric 2  Next hop        AS path
* 0.0.0.0/0          S   5                       >192.168.1.1
* 10.0.0.0/8         S   5                       >192.168.1.2
* 172.16.0.0/12      S 100                        Discard

inet6.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

{master}[edit]
ibm@J08E-re0# run show route protocol static

inet.0: 11 destinations, 13 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[Static/5] 00:00:46
                    > to 192.168.1.1 via ge-1/0/5.0
10.0.0.0/8         *[Static/5] 00:00:46
                    > to 192.168.1.2 via ge-1/0/5.0
172.16.0.0/12      *[Static/100] 00:00:47
                      Discard

inet6.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

{master}[edit]
ibm@J08E-re0# run show route protocol static extensive

inet.0: 11 destinations, 13 routes (11 active, 0 holddown, 0 hidden)
0.0.0.0/0 (1 entry, 1 announced)
TSI:
KRT in-kernel 0.0.0.0/0 -> {192.168.1.1}
        *Static Preference: 5
                Next hop type: Router, Next hop index: 1325
                Next-hop reference count: 3
                Next hop: 192.168.1.1 via ge-1/0/5.0, selected
                State: <Active Int Ext>
                Age: 2:21
                Task: RT
                Announcement bits (1): 0-KRT
                AS path: I

10.0.0.0/8 (1 entry, 1 announced)
TSI:
KRT in-kernel 10.0.0.0/8 -> {192.168.1.2}
        *Static Preference: 5
                Next hop type: Router, Next hop index: 1326
                Next-hop reference count: 3
                Next hop: 192.168.1.2 via ge-1/0/5.0, selected
                State: <Active Int Ext>
```

```
                Age: 2:21
                Task: RT
                Announcement bits (1): 0-KRT
                AS path: I

172.16.0.0/12 (1 entry, 1 announced)
TSI:
KRT in-kernel 172.16.0.0/12 -> {}
        *Static Preference: 100
                Next hop type: Discard
                Next-hop reference count: 2
                State: <Active Int Ext>
                Age: 2:22
                Task: RT
                Announcement bits (1): 0-KRT
                AS path: I

inet6.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)

{master}[edit]
ibm@J08E-re0#
```

10.5 Multicast

IPv4 has the following fundamental types of addresses:

Unicast address Used to send a packet to a single destination.

Broadcast address Used to send a datagram to an entire subnetwork.

Multicast address Used to send a datagram to a set of hosts that can be on different subnetworks and that are configured as members of a multicast group.

A multicast datagram is delivered to destination group members with the same best-effort reliability as a standard unicast IP datagram. Multicast datagrams are not guaranteed to reach all members of a group or to arrive in the same order in which they were transmitted. The only difference between a multicast IP packet and a unicast IP packet is the presence of a group address in the IP header destination address field. Multicast addresses use the Class D address format.

Individual hosts can join or leave a multicast group at any time. The physical location and the number of members in a multicast group have no restrictions.

A host can be a member of more than one multicast group at any time and does not have to belong to a group to send packets to members of a group.

Routers use a group membership protocol to learn about the presence of group members on directly attached subnetworks. When a host joins a multicast group, it transmits a group membership protocol message for the group or groups that it wants to receive. It also sets its IP process and network interface card to receive frames addressed to the multicast group.

The Internet *multicast backbone* (MBone) is an interconnected set of subnetworks and routers that support the delivery of IP multicast traffic. The MBone is a virtual network that is layered on top of sections of the physical Internet. The MBone is composed of islands of multicast routing capability that are connected to other islands by virtual point-to-point links

called *tunnels*. With tunnels, multicast traffic can pass undisturbed through the parts of the Internet that are not multicast capable.

Because the MBone and the Internet have different topologies, multicast routers execute a separate routing protocol to decide how to forward multicast packets.

For more information about multicast, see *JUNOS Software Multicast Protocols Configuration Guide*.

Multicast addresses

Multicast host group addresses are defined to be the IP addresses whose high-order four bits are 1110. This order has an address range of 224.0.0.0–239.255.255.255 or 224.0.0.0/4. (These addresses also are referred to as *Class D addresses*.)

The Internet Assigned Numbers Authority (IANA) maintains a list of registered IP multicast groups. The base address 224.0.0.0 is reserved and cannot be assigned to any group. The block of multicast addresses 224.0.0.1–224.0.0.255 is reserved for local wire use. Groups in this range are assigned for various uses, including routing protocols and local discovery mechanisms.

The range 239.0.0.0–239.255.255.255 is reserved for administratively scoped addresses. Packets addressed to administratively scoped multicast addresses do not cross configured administrative boundaries. Also administratively scoped multicast addresses are locally assigned. Therefore, these addresses do not need to be unique across administrative boundaries.

Overview of multicast snooping

Network devices, such as routers, operate mainly at the packet level or Layer 3. Other network devices, such as bridges or LAN switches, operate mainly at the frame level or Layer 2. Multicasting functions are mainly at the packet level (Layer 3). However, there is a way to map Layer 3 IP multicast group addresses to Layer 2 MAC multicast group addresses at the frame level.

Routers can handle both Layer 2 and Layer 3 addressing information because the frame and its addresses must be processed to access the encapsulated packet inside. Routers can easily run Layer 3 multicast protocols, such as Protocol Independent Multicast (PIM) or Internet Group Management Protocol (IGMP), and determine where to forward multicast content or when a host on an interface joins or leaves a group. However, bridges and LAN switches, as Layer 2 devices, are not supposed to have access to the multicast information inside the packets their frames carry.

How then do bridges and other Layer 2 devices determine when a device on an interface joins or leaves a multicast tree? How do they determine whether a host on an attached LAN is interested in receiving the content of a particular multicast group?

The answer is for the Layer 2 device to implement a series of procedures known as *multicast snooping*. Multicast snooping is a general term and applies to the process of a Layer 2 device “snooping” at the Layer 3 packet content to determine which actions are taken to process or forward a frame. More specific forms of snooping are possible, such as IGMP snooping or PIM snooping. In all cases, snooping involves a device configured to function at Layer 2 having access to normally “forbidden” Layer 3 (packet) information. However, snooping makes multicasting more efficient in these devices.

For details about IGMPv1, IGMPv2, and IGMPv3, see the following standards:

- For IGMPv1, see RFC 1112, Host extensions for IP multicasting at:

  http://www.faqs.org/rfcs/rfc1112.html
- For IGMPv2, see RFC 2236, Internet Group Management Protocol, Version 2, at:

  http://www.faqs.org/rfcs/rfc2236.html
- For IGMPv3, see RFC 3376, Internet Group Management Protocol, Version 3, at:

  http://www.faqs.org/rfcs/rfc3376.html

IGMP snooping configuration

IGMP snooping regulates multicast traffic in a switched network. With IGMP snooping enabled, a LAN switch monitors the IGMP transmissions between a host (a network device) and a multicast router. It tracks the multicast groups and associated member interfaces. The switch uses that information to make intelligent multicast-forwarding decisions and forwards traffic to the intended destination interfaces. IBM j-type Ethernet switches support IGMPv1, IGMPv2, and IGMPv3 (INCLUDE mode only).

IGMP snooping regulates multicast traffic in a switched network. With IGMP snooping enabled, a LAN switch monitors the IGMP transmissions between a host (a network device) and a multicast router, tracking the multicast groups and associated member ports. The switch uses that information to make intelligent multicast-forwarding decisions and forward traffic to the intended destination interfaces.

You can configure IGMP snooping on one or more VLANs to allow the switch to examine IGMP packets and make forwarding decisions based on packet content. By default, IGMP snooping is enabled on IBM j-type Ethernet switches.

To enable IGMP snooping for all VLANs in your network, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols igmp-snooping vlan all
```

To enable IGMP snooping for a specific VLAN in your network, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols igmp-snooping vlan specific_vlan_name
```

You can configure the switch to immediately remove a group membership from an interface when it receives a leave message from that interface without waiting for any other IGMP messages to be exchanged (IGMPv2 only). Use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols igmp-snooping vlan vlan immediate-leave
```

To statically configure IGMP group membership on a port, use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols igmp-snooping vlan vlan interface interface static
group multicast_group
```

To statically configure an interface as a switching interface toward a multicast router (the interface to receive multicast traffic), use the following command:

```
{master}[edit]
ibm@J08E-re0# set protocols igmp-snooping vlan vlan interface interface
multicast-router-interface
```

To verify the operation of IGMP snooping, use the following commands:

```
{master}[edit]
ibm@J08E-re0> show igmp-snooping membership

{master}[edit]
ibm@J08E-re0> show igmp-snooping route

{master}[edit]
ibm@J08E-re0> show igmp-snooping statistics

{master}[edit]
ibm@J08E-re0> show igmp-snooping vlans
```

Example 10-5 shows how to configure and verify IGMP snooping.

Example 10-5 IGMP snooping configuration

```
{master}[edit]
ibm@J08E-re0# set protocols igmp-snooping vlan all

{master}[edit]
ibm@J08E-re0# set protocols igmp-snooping vlan Users immediate-leave

{master}[edit]
ibm@J08E-re0# set protocols igmp-snooping vlan Servers interface xe-3/0/0.0 static
group 230.5.5.5

{master}[edit]
ibm@J08E-re0# show protocols igmp-snooping
vlan all;
vlan Users {
    immediate-leave;
}
vlan Servers {
    interface xe-3/0/0.0 {
        static {
            group 230.5.5.5;
        }
    }
}

{master}[edit]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete

{master}[edit]
ibm@J08E-re0# run show igmp-snooping membership
VLAN: Servers
    230.5.5.5      *
        Interfaces: xe-3/0/0.0
```

```
{master}[edit]
ibm@J08E-re0# run show igmp-snooping membership detail
VLAN: default Tag: 0 (Index: 2)
VLAN: Servers Tag: 200 (Index: 5)
   Group: 230.5.5.5
   Receiver count: 1, Flags: <V2-hosts Static>
     xe-3/0/0.0 Uptime: 00:01:03
VLAN: Users Tag: 100 (Index: 6)
VLAN: Voice Tag: 300 (Index: 7)
VLAN: RV Tag: 500 (Index: 9)

{master}[edit]
ibm@J08E-re0# run show igmp-snooping route
VLAN             Group          Next-hop
default          224.0.0.0, *
Servers          224.0.0.0, *
Servers          230.5.5.5, *       1333
Users            224.0.0.0, *
Voice            224.0.0.0, *
RV               224.0.0.0, *

{master}[edit]
ibm@J08E-re0# run show igmp-snooping statistics
Bad length: 0 Bad checksum: 0 Invalid interface: 0
Not local: 0 Receive unknown: 0 Timed out: 0


    IGMP Type           Received      Transmited    Recv Errors
    Queries:                   0              31              0
    Reports:                   0               0              0
    Leaves:                    0               0              0
    Other:                     0               0              0

{master}[edit]
ibm@J08E-re0# run show igmp-snooping vlans
VLAN                            Interfaces Groups MRouters Receivers RxVlans
RV                                       2      0        0         0       0
Servers                                  4      1        0         0       0
Users                                    4      0        0         0       0
Voice                                    1      0        0         0       0
default                                  1      0        0         0       0

{master}[edit]
ibm@J08E-re0#
```

10.6 Virtual Router Redundancy Protocol

You can configure the Virtual Router Redundancy Protocol (VRRP) on IBM j-type Ethernet switches and routers. When VRRP is configured, the device act as virtual routing platforms. With VRRP, hosts on a LAN can use redundant routing platforms on that LAN without requiring more than the static configuration of a single default route on the hosts.

The VRRP routing platforms share the IP address corresponding to the default route configured on the hosts. At any time, one of the VRRP routing platforms is the master (active), and the others are backups. If the master routing platform fails, one of the backup routing platforms becomes the new master. It provides a virtual default routing platform and enables traffic on the LAN to be routed without relying on a single routing platform. By using VRRP, backup equipment can take over a failed one within a few seconds, which is done with a minimum loss of VRRP traffic and without interaction with the hosts.

VRRP for IPv6 provides a much faster switchover to an alternate default routing platform than IPv6 Neighbor Discovery (ND) procedures. VRRP for IPv6 does not support the authentication-type key.

VRRP dynamically elects master and backup routing platforms. You can also force assignment of master and backup routing platforms using priorities 1–255, with 255 being the highest priority. In VRRP operation, the default master routing platform sends advertisements to back up routing platforms at regular intervals. The default interval is 1 second. If the backup routing platforms do not receive an advertisement for a set period, the backup routing platform with the highest priority takes over as the master and begins forwarding packets.

Priority 255: Priority 255 cannot be set for Routed VLAN Interfaces (RVIs).

VRRP is defined in RFC 3768, Virtual Router Redundancy Protocol.

Configuring VRRP

To create a VRRP group and assign a virtual IP to it, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces interface family inet address ip_address vrrp-group
vrrp_group virtual-address virtual_ip_address
```

The virtual link local address must be on the same subnet as the physical interface address.

You can configure the priority order in which this equipment, functioning as a backup router, becomes the master router if the master router becomes nonoperational. In this case, you configure VRRP priority by using the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces interface family inet address ip_address vrrp-group
vrrp_group priority priority
```

To specify the interval in seconds in which the master router sends advertisement packets to the members of the VRRP group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces interface family inet address ip_address vrrp-group
vrrp_group advertise-interval interval
```

The interval is 1–255 seconds. To specify the interval in milliseconds in which the master router sends advertisement packets to the members of the VRRP group, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces interface family inet address ip_address vrrp-group
vrrp_group fast-interval interval
```

The interval is 100–999 milliseconds.

By default, a higher-priority backup router preempts a lower-priority master router. To explicitly enable the master router to be preempted, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces interface family inet address ip_address vrrp-group
vrrp_group preempt
```

To prohibit a higher-priority backup router from preempting a lower priority master router, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces interface family inet address ip_address vrrp-group
vrrp_group no-preempt
```

To authenticate VRRP advertisements with simple or MD5 authentication, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces interface family inet address ip_address vrrp-group
vrrp_group authentication-type [simple|md5] authentication-key key
```

By default the virtual VRRP address is used only for transit data. To accept data directed to the virtual VRRP address, use the following command:

```
{master}[edit]
ibm@J08E-re0# set interfaces interface family inet address ip_address vrrp-group
vrrp_group accept-data
```

To show the VRRP status, use the following command:

```
{master}
ibm@J08E-re0> show vrrp [ extensive | summary ]
```

Example 10-6 shows the configuration of three VRRP groups with the following characteristics:

- VRRP group 1
  - Virtual IP of 192.168.1.101
  - Accept-data
  - Priority 50
- VRRP group 2
  - Virtual IP of 192.168.1.102
  - No master preemption
  - Authentication simple text with key “auth-key-2”
  - Advertise-interval of 2 seconds
- VRRP group 3
  - Virtual IP of 192.168.1.103
  - Master preemption
  - Authentication simple MD5 with key “auth-key-3”
  - Fast-interval of 200 msec.

Example 10-6 VRRP configuration

```
{master}[edit]
ibm@J08E-re0# edit interfaces ge-1/0/5.0 family inet address 192.168.1.100/24

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 1 virtual-address 192.168.1.101
```

```
{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 1 accept-data

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 1 priority 50

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 2 virtual-address 192.168.1.102

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 2 no-preempt

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 2 authentication-type simple authentication-key
auth-key-2

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 2 advertise-interval 2

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 3 virtual-address 192.168.1.103

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 3 preempt

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 3 authentication-type md5 authentication-key
auth-key-3

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# set vrrp-group 3 fast-interval 200

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# show
vrrp-group 1 {
    virtual-address 192.168.1.101;
    priority 50;
    accept-data;
}
vrrp-group 2 {
    virtual-address 192.168.1.102;
    advertise-interval 2;
    no-preempt;
    authentication-type simple;
    authentication-key "$9$-JVb2ZUHPT324mfznpuRhSr87JZUiqm"; ## SECRET-DATA
}
vrrp-group 3 {
    virtual-address 192.168.1.103;
    fast-interval 200;
    preempt;
    authentication-type md5;
    authentication-key "$9$8/rXxd24ZiqfdbjHm56/uO1RlMY24ZDi"; ## SECRET-DATA
}
```

```
{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# commit
re0:
configuration check succeeds
re1:
commit complete
re0:
commit complete
{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# run show vrrp summary
Interface     State       Group   VR state       VR Mode    Type   Address
ge-1/0/5.0    up            1   backup         Active    lcl    192.168.1.100
                                                         vip    192.168.1.101
ge-1/0/5.0    up            2   master         Active    lcl    192.168.1.100
                                                         vip    192.168.1.102
ge-1/0/5.0    up            3   master         Active    lcl    192.168.1.100
                                                         vip    192.168.1.103

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0# run show vrrp extensive
Interface: ge-1/0/5.0, Interface index :71, Groups: 3, Active :3
  Interface VRRP PDU statistics
    Advertisement sent                      :7189
    Advertisement received                  :280
    Packets received                        :280
    No group match received                 :0
  Interface VRRP PDU error statistics
    Invalid IPAH next type received         :0
    Invalid VRRP TTL value received         :0
    Invalid VRRP version received           :0
    Invalid VRRP PDU type received          :0
    Invalid VRRP authentication type received:0
    Invalid VRRP IP count received          :0
    Invalid VRRP checksum received          :0

Physical interface: ge-1/0/5, Unit: 0, Address: 192.168.1.100/24
  Index: 71, SNMP ifIndex: 531, VRRP-Traps: disabled
  Interface state: up, Group: 1, State: backup, VRRP Mode: Active
  Priority: 50, Advertisement interval: 1, Authentication type: none <----- Auth
  Delay threshold: 100, Computed send rate: 5
  Preempt: yes, Accept-data mode: yes, VIP count: 1, VIP: 192.168.1.101
  Dead timer: 3.194s, Master priority: 100, Master router: 192.168.1.90
  Virtual router uptime: 00:16:56
  Tracking: disabled
  Group VRRP PDU statistics
    Advertisement sent                      :871
    Advertisement received                  :280
  Group VRRP PDU error statistics
    Bad authentication Type received        :0
    Bad password received                   :0
    Bad MD5 digest received                 :0
    Bad advertisement timer received        :0
    Bad VIP count received                  :0
    Bad VIPADDR received                    :0
  Group state transition statistics
```

```
    Idle to master transitions              :0
    Idle to backup transitions              :1
    Backup to master transitions            :1
    Master to backup transitions            :1

Physical interface: ge-1/0/5, Unit: 0, Address: 192.168.1.100/24
  Index: 71, SNMP ifIndex: 531, VRRP-Traps: disabled
  Interface state: up, Group: 2, State: master, VRRP Mode: Active
  Priority: 100, Advertisement interval: 2, Authentication type: simple <--- Auth
  Delay threshold: 100, Computed send rate: 5
  Preempt: no, Accept-data mode: no, VIP count: 1, VIP: 192.168.1.102
  Advertisement Timer: 0.612s, Master router: 192.168.1.100
  Virtual router uptime: 00:16:56, Master router uptime: 00:16:48
  Virtual Mac: 00:00:5e:00:01:02
  Tracking: disabled
  Group VRRP PDU statistics
    Advertisement sent                      :579
    Advertisement received                  :0
  Group VRRP PDU error statistics
    Bad authentication Type received        :0
    Bad password received                   :0
    Bad MD5 digest received                 :0
    Bad advertisement timer received        :0
    Bad VIP count received                  :0
    Bad VIPADDR received                    :0
  Group state transition statistics
    Idle to master transitions              :0
    Idle to backup transitions              :1
    Backup to master transitions            :1
    Master to backup transitions            :0

Physical interface: ge-1/0/5, Unit: 0, Address: 192.168.1.100/24
  Index: 71, SNMP ifIndex: 531, VRRP-Traps: disabled
  Interface state: up, Group: 3, State: master, VRRP Mode: Active
  Priority: 100, Advertisement interval: .200, Authentication type: md5 <--- Auth
  Delay threshold: 100, Computed send rate: 5
  Preempt: yes, Accept-data mode: no, VIP count: 1, VIP: 192.168.1.103
  Advertisement Timer: 0.175s, Master router: 192.168.1.100
  Virtual router uptime: 00:16:56, Master router uptime: 00:16:51
  Virtual Mac: 00:00:5e:00:01:03
  Tracking: disabled
  Group VRRP PDU statistics
    Advertisement sent                      :5739
    Advertisement received                  :0
  Group VRRP PDU error statistics
    Bad authentication Type received        :0
    Bad password received                   :0
    Bad MD5 digest received                 :0
    Bad advertisement timer received        :0
    Bad VIP count received                  :0
    Bad VIPADDR received                    :0
  Group state transition statistics
    Idle to master transitions              :0
    Idle to backup transitions              :1
    Backup to master transitions            :1
```

```
    Master to backup transitions             :0

{master}[edit interfaces ge-1/0/5 unit 0 family inet address 192.168.1.100/24]
ibm@J08E-re0#
```

10.7 More information

All of the following documents are available on the IBM support site at the following web address:

http://www.ibm.com/systems/support/supportsite.wss/brandmain?brandind=5375876

For additional information and step-by-step installation procedures, see the following documentation for your switch:

- *IBM Ethernet Switch J48E Complete Hardware Guide,* GA32-0663
- *IBM Ethernet Switch J48E Quick Start,* GA32-0664
- *IBM Ethernet Switch J08E Complete Hardware Guide,* GA32-0665
- *IBM Ethernet Switch J08E Quick Start,* GA32-0666
- *IBM Ethernet Switch J16E Complete Hardware Guide,* GA32-0667
- *IBM Ethernet Switch J16E Quick Start,* GA32-0668

For more information about the IBM j-type Ethernet routers the related hardware, see the following documentation:

- *IBM j-type m-series Ethernet Router Dense Port Concentrators (DPC) Guide,* GA32-0661
- *IBM j-type m-series Ethernet Routing Engine Installation Instructions,* GA32-0682
- *IBM j-type m-series Ethernet Services PIC Guide,* GA32-0662
- *IBM Ethernet Router J02M Hardware Guide,* GA32-0655
- *IBM Ethernet Router J02M Quick Start,* GA32-0656
- *IBM Ethernet Router J06M Hardware Guide,* GA32-0657
- *IBM Ethernet Router J06M Quick Start,* GA32-0658
- *IBM Ethernet Router J11M Hardware Guide,* GA32-0659
- *IBM Ethernet Router J11M Quick Start,* GA32-0660
- *JUNOS Software IBM j-type m-series Ethernet Routers Solutions Guide,* GA32-0683

For more information about the Junos software, see the following documentation:

- *Juniper Web Device Manager for IBM j-type Ethernet Switches and Routers Interface User Guide,* GA32-0688
- *JUNOS Software Access Privilege Configuration Guide,* GA32-0696
- *JUNOS Software Broadband Subscriber Management Solutions Guide,* GA32-0709
- *JUNOS Software Class of Service Configuration Guide,* GA32-0738
- *JUNOS Software CLI User Guide,* GA32-0697
- *JUNOS Software Configuration and Diagnostic Automation Guide,* GA32-0679
- *JUNOS Software Ethernet Routing Engine Media Upgrade Kit,* GA32-0681
- *JUNOS Software Feature Guide,* GA32-0739

- *JUNOS Software Hierarchy and RFC Reference,* GA32-0712
- *JUNOS Software High Availability Configuration Guide,* GA32-0670
- *JUNOS Software IBM j-type m-series Ethernet Routers Layer 2 Configuration Guide,* GA32-0708
- *JUNOS Software Installation and Upgrade Guide,* GA32-0695
- *JUNOS Software Interfaces Command Reference,* GA32-0672
- *JUNOS Software JUNOScript API Guide,* GA32-0674
- *JUNOS Software MPLS Applications Configuration Guide,* GA32-0702
- *JUNOS Software Multicast Protocols Configuration Guide,* GA32-0703
- *JUNOS Software NETCONF API Guide,* GA32-0678
- *JUNOS Software Network Interfaces Configuration Guide,* GA32-0706
- *JUNOS Software Network Management Configuration Guide,* GA32-0698
- *JUNOS Software Policy Framework Configuration Guide,* GA32-0704
- *JUNOS Software Routing Protocols and Policies Command Reference,* GA32-0673
- *JUNOS Software Services Interfaces Configuration Guide,* GA32-0707
- *JUNOS Software Subscriber Access Configuration Guide,* GA32-0711
- *JUNOS Software System Basics and Services Command Reference,* GA32-0671
- *JUNOS Software System Log Messages Reference,* GA32-0675
- *JUNOS Software VPNs Configuration Guide,* GA32-0705
- *JUNOScope Software User Guide,* GA32-0670
- *JUNOS Software Policy Framework Configuration Guide*, GA32-0704

11

Class of service

Packet networks have evolved to deliver data, voice, and video, the quality of service (QoS). As a result, it has become increasingly important to ensure that applications are afforded the preferential treatment required to operate properly at all times. In general terms, QoS is the ability of a network to provide a defined level of service for particular traffic.

Class of service: The Junos implementation of quality of service is called *class of service* (COS).

The problem with such a broad definition is that it is open. What constitutes service quality? What are the parameters that affect service in a packet-based network? How do you define particular traffic, or, specifically, how do you determine which traffic is given what service?

The first question can be answered by understanding that the following metrics are considered to be standard measurements of service quality:

- End-to-end packet delay
- Delay jitter
- Available capacity
- Drop probability

This chapter includes the following topics:

- Overview of class of service
- Junos class-of-service components
- Hardware limitations
- Packet flow
- Packet classification
- Forwarding classes
- Policing
- Rewrite rules
- Packet loss priority
- Schedulers
- RED profiles
- Tail drop profiles
- Case study

© Copyright IBM Corp. 2011. All rights reserved.

11.1 Overview of class of service

When a network experiences congestion and delay, some packets must be dropped. The Junos implementation of quality of service is called *class of service* (COS). With COS, you can divide traffic into classes and offer various levels of throughput and packet loss when congestion occurs. Packet loss can happen according to rules that you configure.

For interfaces that carry IPv4, IPv6, and Multiprotocol Label Switching (MPLS) traffic, you can configure Junos class-of-service features to provide multiple classes of service for different applications. On the router, you can configure multiple forwarding classes for transmitting packets and define which packets are placed into each output queue. You can also schedule the transmission service level for each queue and manage congestion by using a Random Early Detection (RED) algorithm.

The Junos class-of-service features provide a set of mechanisms that you can use to provide differentiated services when best-effort traffic delivery is insufficient. In designing class-of-service applications, give careful consideration to your service needs. Thoroughly plan and design your class-of-service configuration to ensure consistency across all routers in a class-of-service domain. You must also consider all the routers and other networking equipment in the class-of-service domain to ensure interoperability among all equipment.

Because IBM routers implement COS in hardware rather than in software, you can experiment with and deploy class-of-service features without adversely affecting packet forwarding and routing performance

This chapter explains the steps to configure the following class-of-service features on the IBM j-type series and provides examples of each. You must be familiar with the following basic of QoS topics:

- Packet classification
- Forwarding classes
- Policing
- Rewrite rules
- Packet loss priority
- Schedulers
- Drop profiles

11.2 Junos class-of-service components

COS works by examining traffic entering at the edge of your network. The edge routers classify traffic into defined service groups, to provide the special treatment of traffic across the network. For example, voice traffic can be sent across certain links, and data traffic can use other links. In addition, the data traffic streams can be serviced differently along the network path to ensure that higher-paying customers receive better service. As the traffic leaves the network at the far edge, you can reclassify the traffic.

To support COS, you must configure each router in the network. Generally, each router examines the packets that enter it to determine their class-of-service settings. These settings then dictate which packets are first transmitted to the next downstream router. In addition, the routers at the edges of the network might be required to alter the class-of-service settings of the packets that enter the network from customer or peer networks.

Junos class-of-service configuration is based on the following components:

- Packet classification

  Packet classification associates incoming packets with a particular class-of-service servicing level. In Junos software, classifiers associate packets with a forwarding class and loss priority and assign packets to output queues based on the associated forwarding class. Junos software supports two general types of classifiers:

  - Behavior aggregate
  - Multifield traffic classifiers

- Forwarding classes

  Forwarding classes group the packets for transmission. Based on forwarding classes, you assign packets to output queues. Forwarding classes affect the forwarding, scheduling, and marking policies applied to packets as they transit a switching platform. By default, the following categories of forwarding classes are defined:

  - Best effort
  - Assured forwarding
  - Expedited forwarding
  - Network control

- Policers

  Policers limit traffic of a certain class to a specified bandwidth and burst size. Packets that exceed the policer limits can be discarded. You define policers with filters that can be associated with input interfaces.

- Rewrite rules

  A rewrite rule sets the appropriate class-of-service bits in the outgoing packet. It allows the next downstream device to classify the packet into the appropriate service group. Rewriting, or marking, outbound packets is useful when the switch is at the border of a network and must alter the class-of-service values to meet the policies of the targeted peer.

- Scheduler

  Each interface has multiple queues assigned to store packets. The switch determines which queue to service based on a particular method of scheduling. This process often involves determining which type of packet to transmit before another. You can define the priority, bandwidth, delay buffer size, and drop profiles to be applied to a particular queue for packet transmission.

  A scheduler map associates a specified forwarding class with a scheduler configuration.

- Drop profiles and loss priority

  A *drop profile* is a mechanism that defines parameters that allow packets to be dropped from the network. Drop profiles define the meanings of the loss priorities. When you configure drop profiles, you are essentially setting the value for queue fullness. The queue fullness represents a percentage of the queue used to store packets in relation to the total amount that has been allocated for that specific queue.

  Loss priorities set the priority of dropping a packet. Loss priority affects the scheduling of a packet without affecting the packet's relative ordering. You can use the loss priority setting to identify packets that have experienced congestion. Typically you mark packets that exceed a service level with a high loss priority.

11.3 Hardware limitations

Generally, the Layer 3 class-of-service hardware capabilities and limitations for IBM j-type m-series Ethernet routers are the same as for M-series Multiservice Edge Routers (M120 routers in particular).

The following scaling and performance parameters apply to IBM j-type m-series Ethernet routers:

- Eight classifiers
- Eight rewrite tables
- Eight queues per port
- 32 weighted RED (WRED) profiles
- 100 ms queue buffering
- Line-rate class-of-service features

11.4 Packet flow

The class-of-service architecture for IBM j-type m-series Ethernet routers, such as the IBM Ethernet Router J11M, is similar in concept to, but different from, other routers. Figure 11-1 shows the general architecture for IBM j-type m-series Ethernet routers.

Figure 11-1 IBM j-type m-series packet flow

The IBM j-type m-series Ethernet router can classify incoming packets at the ingress dense port concentrator. Fixed classification places all packets in the same forwarding class, or the usual multifield (MF) or behavior aggregate (BA) classifications can be used to treat packets differently. A BA classification with firewall filters can be used for classification based on IP precedence, Differentiated Services Code Point (DSCP), IEEE, or other bits in the frame or packet header.

However, the IBM j-type m-series Ethernet router can also employ multiple BA classifiers on the same logical interface. The logical interfaces do not have to employ the same type of BA classifier. For example, a single logical interface can use classifiers based on IP precedence and IEEE 802.1p. If the class-of-service bits of interest are on the inner VLAN tag of a dual-tagged VLAN interface, the classifier can examine either the inner or outer bits. (By default, the classification is done based on the outer VLAN tag.)

Internal fabric scheduling in the IBM j-type m-series Ethernet router is based on only two queues: high priority and low priority. Strict high-priority queuing is also supported in the high-priority category.

Egress port scheduling supports up to eight queues per port using a form of round-robin queue servicing. The supported priority levels are strict-high, high, medium-high, medium-low, and low. The IBM j-type m-series Ethernet router architecture supports both early discard and tail drop on the queues.

For more information about this topic, see *Class of Service Configuration Guide*, GA32-0738.

11.5 Packet classification

QoS proposals have been standardized by the Internet Engineering Task Force (IETF). In particular, Junos software adheres to the differentiated services model (based on RFC 2475). Packets are normally classified at the edge of the network by marking the DSCP field in the IP header, which defines the class of service for that packet as it moves through the network.

As packets are classified in Junos software, they are assigned to a forwarding class that specifies both the transmit queue and the packet loss priority. The per-hop behavior of a packet is determined by its specific queue and loss priority. Later, this book explains how traffic classes are specified and associated with queues. It also explains how different loss priorities affect the drop probability of a packet.

The following general types of classifiers are possible:

- Behavior aggregate classifiers
- Multifield classifiers

For a specified interface, you can configure both an MF classifier and a BA classifier without conflicts. In such cases, BA classification is performed first, followed by MF classification. If the two classification results conflict, the MF classification result overrides the BA classification result.

11.5.1 Behavior aggregate classification

The BA classifier maps a class-of-service value to a forwarding class and loss priority. The forwarding class determines the output queue. The loss priority is used by schedulers with the RED algorithm to control packet discard during periods of congestion. See 11.11, “RED profiles” on page 356.

The types of BA classifiers are based on which part of the incoming packet the classifier examines:

- DSCP for IP DiffServ
- DSCP for IPv6 DiffServ
- IP precedence bits
- MPLS EXP bits
- IEEE 802.1p class-of-service bits
- IEEE 802.1ad drop eligible indicator (DEI) bit

Default classification

The software automatically assigns an implicit default IP precedence classifier to all logical interfaces. If you enable the MPLS protocol family on a logical interface, a default MPLS EXP classifier is automatically applied to that logical interface.

Other default classifiers (such as classifiers for IEEE 802.1p bits and DSCP) require you to explicitly associate a default classification table with a logical interface. Table 11-1 shows the list of default classifiers.

Table 11-1 Default classifiers

Default classifier type	Default classifier name
DSCP	dscp-default
DSCP for IPv6	dscp-ipv6-default
MPLS experimental code point	exp-default
IEEE-802.1 code point	ieee8021p-default
IEEE-802.1ad (DEI) code point	ieee8021ad-default
IPv4 precedence code point	ipprec-default

The default classifiers are listed by typing the command shown in Example 11-1.

Example 11-1 DSCP default classifiers

```
ibm@J11M-re0> show class-of-service classifier name dscp-default
Classifier: dscp-default, Code point type: dscp, Index: 7
  Code point         Forwarding class                    Loss priority
  000000             best-effort                         low
  000001             best-effort                         low
  000010             best-effort                         low
  000011             best-effort                         low
  000100             best-effort                         low
  000101             best-effort                         low
  000110             best-effort                         low
  000111             best-effort                         low
  001000             best-effort                         low
  001001             best-effort                         low
  001010             assured-forwarding                  low
  001011             best-effort                         low
  001100             assured-forwarding                  high
  001101             best-effort                         low
  001110             assured-forwarding                  high
  001111             best-effort                         low
  010000             best-effort                         low
  010001             best-effort                         low
  010010             best-effort                         low
  010011             best-effort                         low
  010100             best-effort                         low
  010101             best-effort                         low
  010110             best-effort                         low
  010111             best-effort                         low
  011000             best-effort                         low
  011001             best-effort                         low
  011010             best-effort                         low
```

```
  011011              best-effort                          low
  011100              best-effort                          low
  011101              best-effort                          low
  011110              best-effort                          low
  011111              best-effort                          low
  100000              best-effort                          low
  100001              best-effort                          low
  100010              best-effort                          low
  100011              best-effort                          low
  100100              best-effort                          low
  100101              best-effort                          low
  100110              best-effort                          low
  100111              best-effort                          low
  101000              best-effort                          low
  101001              best-effort                          low
  101010              best-effort                          low
  101011              best-effort                          low
  101100              best-effort                          low
  101101              best-effort                          low
  101110              expedited-forwarding                 low
  101111              best-effort                          low
  110000              network-control                      low
  110001              best-effort                          low
  110010              best-effort                          low
  110011              best-effort                          low
  110100              best-effort                          low
  110101              best-effort                          low
  110110              best-effort                          low
  110111              best-effort                          low
  111000              network-control                      low
  111001              best-effort                          low
  111010              best-effort                          low
  111011              best-effort                          low
  111100              best-effort                          low
  111101              best-effort                          low
  111110              best-effort                          low
  111111              best-effort                          low

{master}
ibm@J11M-re0>
```

Classifier configuration

You can override the default IP precedence classifier by defining a classifier and applying it to a logical interface. For each protocol family, classifiers are configured under `[class-of-service]` as shown in Example 11-2 on page 340.

A classifier takes a specified bit pattern and attempts to match it to the packet arriving on the interface. If the information in the packet's header matches the specified pattern, the packet is sent to the appropriate queue, defined by the forwarding class associated with the classifier.

Example 11-2 Classifier definition

```
[edit class-of-service]
   classifiers {
                (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence)
                classifier-name {
                        import [classifier-name | default];
                             forwarding-class class-name {
                                          loss-priority level {
                                        code-points [ aliases ] [
                                        bit-patterns ];
                                          }
                             }
                }
   }
```

For a complete description of the command, type the following command:

```
ibm@J11M-re0> help topic class-of-service classifiers
```

You can apply the classifier to an interface as shown in Example 11-3.

Example 11-3 Applying a classifier

```
class-of-service {
    interface interface-name {
        unit unit-number {
            classifiers {
                dscp|dscp-ipv6|exp|inet-precedence classifier-name;
            }
        }
    }
}
```

Classifier example

Example 11-4 defines an IP Precedence Classifier named TEST-CLASSIFIER and associates code points 001, 010, and 011 to forward class Assured Forwarding with low loss priority. Then it verifies the configuration and applies the classifier to the ge-0/0/0 interface.

Example 11-4 Classifier configuration

```
{master}[edit class-of-service]
ibm@J11M-re0# set classifiers inet-precedence TEST-CLASSIFIER forwarding-class
assured-forwarding loss-priority low code-points 001 code-points 010 code-points
011

{master}[edit class-of-service]
ibm@J11M-re0# commit
commit complete

{master}[edit class-of-service]
ibm@J11M-re0# run show class-of-service classifier name TEST-CLASSIFIER
Classifier: TEST-CLASSIFIER, Code point type: inet-precedence, Index: 28143
  Code point         Forwarding class                    Loss priority
  001                assured-forwarding                  low
  010                assured-forwarding                  low
  011                assured-forwarding                  low
```

```
{master}[edit class-of-service]
ibm@J11M-re0# set interfaces ge-0/0/0 unit 0 classifiers inet-precedence
TEST-CLASSIFIER

{master}[edit class-of-service]
ibm@J11M-re0# commit

{master}[edit]
ibm@J11M-re0# run show class-of-service interface ge-0/0/0
Physical interface: ge-0/0/0, Index: 132
Queues supported: 8, Queues in use: 4
  Scheduler map: <default>, Index: 2

  Logical interface: ge-0/0/0.0, Index: 83
    Object                  Name                    Type                    Index
    Classifier              TEST-CLASSIFIER         ip                      28143
```

11.5.2 Code point alias

BA classifiers use class-of-service values such as DSCPs, DSCP IPv6, IP precedence, IEEE 802.1 and MPLS experimental (EXP) bits to associate incoming packets with a particular class-of-service level. You can assign a meaningful name or alias to the class-of-service values and use this alias instead of bits when configuring class-of-service components. These aliases are not part of the specifications but are well-known through usage. For example, the alias for DSCP 101110 is widely accepted as EF (expedited forwarding).

When you configure classes and define classifiers, you can refer to the markers by alias names. You can configure user-defined classifiers in terms of alias names. If the value of an alias changes, it alters the behavior of any classifier that references it.

Default aliases

Example 11-5 shows the default mappings between DSCP bit values and standard aliases. You can see the default mappings for other bit codes (IP Precedence, IEEE 802.1p, MPLS EXP, and so on) issuing the same command.

Example 11-5 DSCP code point alias

```
{master}[edit]
ibm@J11M-re0> show class-of-service code-point-aliases dscp
Code point type: dscp
  Alias              Bit pattern
  af11               001010
  af12               001100
  af13               001110
  af21               010010
  af22               010100
  af23               010110
  af31               011010
  af32               011100
  af33               011110
  af41               100010
  af42               100100
  af43               100110
```

```
  be                  000000
  cs1                 001000
  cs2                 010000
  cs3                 011000
  cs4                 100000
  cs5                 101000
  cs6                 110000
  cs7                 111000
  ef                  101110
  nc1                 110000
  nc2                 111000
```

Configuration

To configure class-of-service code point aliases, include the code-point aliases statement at the [`edit class-of-service`] hierarchy level as shown in Example 11-6.

Example 11-6 Defining code point alias

```
code-point-aliases {
  (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) {
    alias-name bits;
  }
}
```

Example

Example 11-7 shows generating an alias named ea for DSCP code point 001010.

Example 11-7 Configuring a code-point alias

```
{master}[edit]
ibm@J11M-re0# set class-of-service code-point-aliases dscp ea 001010

{master}[edit]
ibm@J11M-re0# commit

{master}[edit]
ibm@J11M-re0# run show class-of-service code-point-aliases dscp
Code point type: dscp
  Alias               Bit pattern
  af11                001010
  af12                001100
  af13                001110
  af21                010010
  af22                010100
  af23                010110
  af31                011010
  af32                011100
  af33                011110
  af41                100010
  af42                100100
  af43                100110
  be                  000000
  cs1                 001000
  cs2                 010000
  cs3                 011000
```

```
cs4                 100000
cs5                 101000
cs6                 110000
cs7                 111000
ea                  001010
ef                  101110
nc1                 110000
nc2                 111000
```

11.5.3 Multifield classification

A MF classifier provides filtering functionality that scans through various packet fields to determine the forwarding class for a packet. Typically, a classifier performs matching operations on the selected fields against a configured value.

An MF classifier typically matches one or more of the six packet header fields:

- Destination address
- Source address
- IP protocol
- Source port
- Destination port
- DSCP

MF classifiers are used when a simple BA classifier is insufficient to classify a packet.

Configuration

The configuration of an MF classifier requires the definition of a firewall filter, which is under the `[firewall family inet]` hierarchy as shown in Example 11-8. The 802.1p bits cannot be used as matching conditions. By the time the packet is processed by the ingress/egress filters, the Ethernet frame has been stripped.

Example 11-8 Firewall filter definition for MF configuration

```
firewall {
                family inet{
                             filter filter-name {
                                          term term-name {
                                             from {
                                        match-conditions;
                                             }
                                             then {
                                        forwarding-class class-name;
                                        loss-priority priority;
                                             }
                                          }
                             }
                }
   }
```

For a complete description of the command, type the following command:

```
ibm@J11M-re0# help topic firewall filter
```

Filters can be applied to logical interfaces both on ingress and egress, as shown in Example 11-9.

Example 11-9 Applying filters

```
interfaces interface-name{
        unit unit-number {
            family inet {
                filter {
                    input filter-name;
                    output filter-name;
                }
            }
        }
    }
```

Example

Example 11-10 classifies traffic with a source address 10.0.0.0/24 and assigns it to forwarding class Assured Forwarding with medium-low loss priority. Then it applies the filter to the ge-0/0/0 interface.

Example 11-10 MF configuration

```
{master}[edit]
ibm@J11M-re0# set firewall family inet filter CoS-TEST term SOURCE from
source-address 10.0.0.0/24

{master}[edit]
ibm@J11M-re0# set firewall family inet filter CoS-TEST term SOURCE then
forwarding-class assured-forwarding

{master}[edit]
ibm@J11M-re0# set firewall family inet filter CoS-TEST term SOURCE then
loss-priority medium-low

{master}[edit]
ibm@J11M-re0# set interfaces ge-0/0/0 unit 0 family inet filter input CoS-TEST

{master}[edit]
ibm@J11M-re0# show firewall family inet filter CoS-TEST
term SOURCE {
    from {
        source-address {
            10.0.0.0/24;
        }
    }
    then {
        loss-priority medium-low;
        forwarding-class assured-forwarding;
    }
}

{master}[edit]
ibm@J11M-re0# show interfaces ge-0/0/0
unit 0 {
    family inet {
```

```
        filter {
            input CoS-TEST;
        }
    }
}
```

11.6 Forwarding classes

You can think of forwarding classes as output queues. In effect, the result of classification is the identification of an output queue for a particular packet. For a classifier to assign an output queue to each packet, it must associate the packet with one of the following forwarding classes:

Expedited forwarding (EF) Provides a low loss, low latency, low jitter, assured bandwidth, end-to-end service.

Assured forwarding (AF) Provides a group of values that you can define and includes four subclasses, AF1, AF2, AF3, and AF4, each with three drop probabilities: low, medium, and high.

Best effort (BE) Provides no service profile. Loss priority is typically not carried in a class-of-service value, and RED drop profiles are more aggressive.

Network Control (NC) Is typically high priority because it supports protocol control.

Default forwarding classes

By default, four queues are assigned to four forwarding classes, each with a queue number, name, and abbreviation. Table 11-2 shows the default definition of the classes. You can rename the forwarding classes associated with the queues supported on your hardware.

Table 11-2 Default forwarding classes

Queue	Forwarding class name	Description
Queue 0	Best effort	The software does not apply any special class-of-service handling to packets with 000000 in the DiffServ field, which is a feature compatible with an earlier version. These packets are usually dropped under congested network conditions.
Queue 1	Expedited forwarding	The software delivers assured bandwidth, low loss, low delay, and low delay variation (jitter) end-to-end for packets in this service class. Routers accept excess traffic in this class, but in contrast to assured forwarding, out-of-profile expedited-forwarding packets can be forwarded out of sequence or dropped.
Queue 2	Assured forwarding	The software offers a high level of assurance that the packets are delivered while the packet flow from the customer stays within a certain service profile that you define. The software accepts excess traffic, but applies a RED drop profile to determine if the excess packets are dropped and not forwarded. Depending on router type, up to four drop probabilities (low, medium-low, medium-high, and high) are defined for this service class.

Queue	Forwarding class name	Description
Queue 3	Network control	The software delivers packets in this service class with a low priority. (These packets are not delay sensitive.) Typically, these packets represent routing protocol hello or keepalive messages. Because loss of these packets jeopardizes proper network operation, delay is preferable to discard.

Configuration

You assign each forwarding class to an internal queue number by including the `forwarding-classes` statement at the `[edit class-of-service]` hierarchy level as shown in Example 11-11.

Example 11-11 Forwarding class definition

```
[edit class-of-service]
forwarding-classes {
   queue queue-number class-name;
}
```

In addition to BA and MF classification, the forwarding class of a packet can be directly determined by the logical interface that receives the packet. This forwarding class of a packet can be configured by using CLI commands as shown in Example 11-12. If configured, this forwarding class overrides the forwarding class from any BA classification that was previously performed on the logical interface.

Example 11-12 Direct forwarding class assignment

```
class-of-service {
    interfaces {
        interface-name {
            unit unit-number{
                forwarding-class class-name;
            }
        }
    }
}
```

11.7 Policing

Policing refers to the ability of a router to measure data rates and, based on this measurement, to either drop or reclassify the traffic. After MF classification is performed, you can instruct a router to measure the rate of the traffic matching the classifier. Then you can either drop or change the forwarding class, or drop the priority of the packet if the measured rate exceeds a configurable threshold.

In simple terms, policers allow the establishment of a data rate, which, if exceeded, results in traffic being either reclassified or dropped. To measure traffic rates, you must determine a measurement interval (or burst limits). Traffic always egresses an interface at line rate. To send traffic at a “lower speed,” bursts must be followed by idle periods, resulting in an average transmit rate lower than the line rate.

The IBM j-type family supports the following types of policers:

- Two-color marking

  A two-color policer meters the traffic stream and classifies packets into two categories of packet loss priority (PLP) according to a configured bandwidth and burst-size limit. You can mark packets that exceed the bandwidth and burst-size limit in some way or discard them.

- Single-rate tricolor marking

  This type of policer, defined in RFC 2697, meters traffic based on the configured Committed Information Rate (CIR), Committed Burst Size (CBS), and the Excess Burst Size (EBS). Traffic is marked as belonging to one of three categories (green, yellow, or red) based on whether the packets arriving are below the CBS (green), exceed the CBS (yellow) but not the EBS, or exceed the EBS (red).

- Two-rate tricolor marking

  This type of policer, which is defined in RFC 2698, meters traffic based on the configured CIR and Peak Information Rate (PIR), along with their associated burst sizes, the CBS, and peak burst size (PBS). Traffic is marked as belonging to one of three categories: green, yellow, or red. It is based on whether the packets arriving are below the CIR (green), exceed the CIR (yellow) but not the PIR, or exceed the PIR (red).

IBM j-type m-series and e-series support: IBM j-type m-series supports two-color marking and two-rate two-color marking. IBM j-type e-series supports only two-color marking.

11.7.1 Two-color marking configuration

The two-color marking configuration uses firewall filters similar to those filters used for packet classification. The first step is to define a policer under the `[firewall policers]` hierarchy as shown in Example 11-13. This policer allows the specification of a generic policer that can be applied to either single or multiple filters.

Example 11-13 Firewall policer definition

```
[edit firewall]
   policer policer-name {
                if-exceeding {
                        bandwidth-limit bps;
                        bandwidth-percent number;
                        burst-size-limit bytes;
                }
                then {
                   discard;
                   forwarding-class forwarding-class-name;
                   loss-priority priority;
                }
   }
```

For a complete description of the command, type the following command:

```
ibm@J11M-re0# help topic firewall policer
```

IBM j-type m-series Ethernet routers and e-series Ethernet switches: Policers can also be applied to a logical interface on IBM j-type m-series Ethernet routers. In this case, all ingress or egress traffic (depending on the policer direction) is policed. Discard is the only supported policer action on IBM j-type e-series Ethernet switches.

A policer specifies a maximum bandwidth, a max-burst size, and an action. The suggested burst limit value of the policer for a low-speed interface is 10 times the interface maximum transmission unit (MTU). For a high-speed interface, the suggested burst size is the transmit rate of the interface times 3–5 milliseconds.

For interface-based policers, the maximum bandwidth can either be specified as an absolute value or as a percentage of the ingress or egress interface line rate. Filter-based policers can only specify the bandwidth in bps, because the transmit data rate it is unknown until the egress interface is determined.

Policers can be applied to a logical interface (only on IBM j-type m-series Ethernet routers) as shown in Example 11-14.

Example 11-14 Policer applied to an interface

```
interfaces {
    interface-name {
        unit unit-number {
            family inet {
                policer input policer-name;
                policer output policer-name
                }
            }
        }
    }
```

Policers can also be applied to a logical interface on IBM j-type m-series Ethernet routers as an action in a firewall filter as shown in Example 11-15.

Example 11-15 Policer applied to a firewall filter

```
firewall {
     family inet {
         filter filter-name {
             term term-name {
                 from {
                       match conditions;
                 }
                 then {
                     policer policer-name;
                     other actions;
                 }
             }
         }
     }
}
```

By using the next term action on a filter doing policing, you can stack one or more policers. You can do this whenever it is desired to limit aggregated traffic from multiple sources, interfaces from exceeding a particular bandwidth, or both.

11.7.2 Two-color marking example

Example 11-16 shows the configuration of a policer, named TEST-POLICER, which limits the output traffic to 2 Mbps on the ge-0/0/0.0 interface on an IBM j-type m-series Ethernet router.

Example 11-16 Two-color marking

```
{master}[edit]
ibm@J11M-re0# set firewall policer TEST-POLICER if-exceeding bandwidth-limit
2000000 burst-size-limit 375k

{master}[edit]
ibm@J11M-re0# set firewall policer TEST-POLICER then discard

{master}[edit]
ibm@J11M-re0# show firewall policer TEST-POLICER
if-exceeding {
    bandwidth-limit 2m;
    burst-size-limit 375k;
}
then discard;

ibm@J11M-re0# set interfaces ge-0/0/0 unit 0 family inet policer output
TEST-POLICER

{master}[edit]
ibm@J11M-re0# show interfaces ge-0/0/0
unit 0 {
    family inet {
        policer {
            output TEST-POLICER;
        }
    }
}

{master}[edit]
ibm@J11M-re0# commit

{master}[edit]
ibm@J11M-re0# run show policer TEST-POLICER-ge-0/0/0.0-inet-o
Policers:
Name                                            Packets
TEST-POLICER-ge-0/0/0.0-inet-o                        0
```

11.8 Rewrite rules

As packets enter or exit a network, edge routers might be required to alter the class-of-service settings of the packets. Rewrite rules set the value of the class-of-service bits within the header of the packet. Each rewrite rule reads the current forwarding class and loss priority information associated with the packet. Then it locates the chosen class-of-service value from a table and writes this class-of-service value into the packet header.

You configure rewrite rules to alter class-of-service values in outgoing packets on the outbound interfaces of an edge router to meet the policies of a targeted peer. This way the

downstream router in a neighboring network can classify each packet into the appropriate service group.

In addition, you often must rewrite a given marker (IP precedence, DSCP, IEEE 802.1p, or MPLS EXP settings) at the inbound interfaces of an edge router to accommodate BA classification by core devices.

11.8.1 Default rewrite rules

By default, rewrite rules are not usually applied to interfaces. The exceptions are MPLS interfaces. All MPLS-enabled interfaces use the default EXP rewrite rule, even if it is not configured. Except for MPLS interfaces, if you want to apply a rewrite rule, you can design your own rule and apply it to an interface. Alternatively, you can apply a default rewrite rule. Table 11-3 shows the default rewrite rules.

Table 11-3 Default rewrite rules

Default rewrite rule type	Default rewrite rule name
DSCP	dscp-default
DSCP for IPv6	dscp-ipv6-default
MPLS experimental code point	exp-default
IEEE-802.1 code point	ieee8021p-default
IEEE-802.1ad (DEI) code point	ieee8021ad-default
IPv4 precedence code point	ipprec-default

To see the contents of these tables, use the command shown in Example 11-17.

Example 11-17 DSCP default rewrite rule

```
{master}[edit]
ibm@J11M-re0# run show class-of-service rewrite-rule name dscp-default
Rewrite rule: dscp-default, Code point type: dscp, Index: 31
  Forwarding class                    Loss priority       Code point
  best-effort                         low                 000000
  best-effort                         high                000000
  expedited-forwarding                low                 101110
  expedited-forwarding                high                101110
  assured-forwarding                  low                 001010
  assured-forwarding                  high                001100
  network-control                     low                 110000
  network-control                     high                111000
```

11.8.2 Configuring rewrite rules

To configure a rewrite rule mapping and associate it with the appropriate forwarding class and code-point alias or bit set, include the `rewrite-rules` statement at the `[edit class-of-service]` hierarchy level as shown in Example 11-18.

Example 11-18 Configuring rewrite rules

```
[edit class-of-service]
   rewrite-rules {
                (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence)
                rewrite-name {
                        import (rewrite-name | default);
                             forwarding-class class-name {
                                     loss-priority level code-point (alias |
                                     bits);
                             }
                }
   }
```

For a complete description of the command, type the following command:

```
ibm@J11M-re0# help topic class-of-service rewrite-rules
```

Example 11-19 shows how to apply a rewrite rule to a logical interface.

Example 11-19 Applying rewrite rules to logical interfaces

```
[edit class-of-service interfaces interface-name unit logical unit-number]
   rewrite-rules {
           dscp (rewrite-name | default) protocol protocol-types;
           dscp-ipv6 (rewrite-name | default);
           exp (rewrite-name | default) protocol protocol-types;
           exp-push-push-push default;
           exp-swap-push-push default;
           ieee-802.1 (rewrite-name | default) inet-prec vlan-tag (outer |
           outer-and-inner);
           inet-precedence (rewrite-name | default) protocol protocol-types;
   }
```

11.8.3 Example of a default rewrite rule

Example 11-20 shows how to apply the default DSCP rewrite rule to ge-0/0/0 interface unit 0.

Example 11-20 Applying the DSCP default rewrite rule

```
{master}[edit]
ibm@J11M-re0# set class-of-service interfaces ge-0/0/0 unit 0 rewrite-rules dscp
default

{master}[edit]
ibm@J11M-re0# commit

{master}[edit]
ibm@J11M-re0# show class-of-service interfaces ge-0/0/0
unit 0 {
```

```
    rewrite-rules {
        dscp default;
    }
}

{master}[edit]
ibm@J11M-re0# run show class-of-service interface ge-0/0/0
Physical interface: ge-0/0/0, Index: 132
Queues supported: 8, Queues in use: 4
  Scheduler map: <default>, Index: 2

  Logical interface: ge-0/0/0.0, Index: 83
    Object                  Name                    Type                    Index
    Rewrite                 dscp-default            dscp                       31
    Classifier              ipprec-compatibility    ip                         13
```

11.9 Packet loss priority

Junos software permits the tagging of packets with a loss priority indicator. This tag performs similar functions as the Discard Eligibility bit in Frame Relay or Cell Loss Priority bit in ATM.

Internally, PLP bits cannot be viewed directly in the device. Externally, PLP bits are carried by using specific bits in the IP precedence, MPLS EXP, or DSCP fields. Default DSCP and IP precedence rewrite and classification tables do not support external communication of PLP for the BE and EF classes.

You can use the loss priority setting to identify packets that have experienced congestion. Typically you mark packets exceeding some service level with a high loss priority. You set loss priority by configuring a classifier or a policer. The loss priority is used later in the workflow to select one of the drop profiles used by RED, as shown in 11.11, “RED profiles” on page 356.

11.10 Schedulers

After traffic is classified, queuing and scheduling can be used to provide different levels of service for the classified traffic. In Junos software, forwarding classes are associated with queues.

You use schedulers to define the properties of output queues. These properties include the amount of interface bandwidth assigned to the queue, the size of the memory buffer allocated for storing packets, the priority of the queue, and the RED drop profiles associated with the queue. You associate the schedulers with forwarding classes with scheduler maps. You can then associate each scheduler map with an interface, configuring the hardware queues, packet schedulers, and RED processes that operate according to this mapping.

11.10.1 Default schedulers

Only two forwarding classes (queues) have default scheduler configuration:

- Best Effort (queue 0) receives 95% of the bandwidth.
- Network Control (queue 3) receives 5% of the bandwidth.

The default drop profile causes the buffer to fill and then discard all packets until it has space.

By default, the Expedited Forwarding and Assured Forwarding classes have no schedulers. You can manually configure resource for these classes. Also by default, each queue can exceed the assigned bandwidth if additional bandwidth is available.

11.10.2 Configuring schedulers

The following terms describe the parameters that are required to configure schedulers:

- Transmission rate

  The transmission-rate control determines the actual traffic bandwidth from each forwarding class that you configure. The rate is specified in bits per second. Each queue is allocated a portion of the bandwidth of the outgoing interface.

  This bandwidth amount can be a fixed value, such as 1 megabit per second (Mbps), a percentage of the total available bandwidth, or the rest of the available bandwidth. You can allow transmission bandwidth to exceed the configured rate if additional bandwidth is available from other queues. In case of congestion, the configured amount of transmission rate is guaranteed for the queue. With this property, you can ensure that each queue receives the amount of bandwidth appropriate to its level of service.

- Scheduler buffer size

  To control congestion at the output stage, you can configure the delay-buffer bandwidth. The delay-buffer bandwidth provides packet buffer space to absorb burst traffic up to the specified duration of delay. When the specified delay buffer becomes full, packets with 100 percent drop probability are dropped from the tail of the buffer.

  For each scheduler, you can configure the buffer size as a percentage of the total buffer or the remaining buffer that is available. The remainder is the buffer percentage that is not assigned to other queues.

- Shaping rate

  Shaping rates control the maximum rate of traffic transmitted on an interface. You can configure the shaping rate so that the interface transmits less traffic than it is physically capable of carrying.

  You can configure shaping rates on logical interfaces. By default, output scheduling is not enabled on logical interfaces. With logical interface scheduling (also called *per-unit scheduling*), you can enable multiple output queues on a logical interface and associate an output scheduler and shaping rate with the queues.

- Priority scheduling

  Priority scheduling determines the order in which an output interface transmits traffic from the queues. It ensures that queues containing important traffic are provided better access to the outgoing interface.

  Priority scheduling is accomplished through a procedure in which the scheduler examines the priority of the queue. Higher priority queues transmit packets ahead of lower priority queues provided that the higher priority forwarding classes retain enough bandwidth credit.

  IBM j-type m-series Ethernet routers support the following priority levels:

  - Low
  - Medium-low
  - Medium-high
  - High
  - Strict-high

IBM j-type e-series Ethernet switches support the following priority levels:

- Low
- Strict-high

For more information about Priority Scheduling on IBM j-type m-series, see “Priority Scheduling Overview” in the *Class of Service Configuration Guide,* GA32-0738.

- Scheduler drop-profile maps

  Drop-profile maps associate drop profiles with a scheduler. Drop-profile map sets the drop profile for a specific PLP and protocol type. The inputs for the drop-profile map are the PLP and the protocol type. The output is the drop profile. The scheduler drop profile defines the drop probability for the RED process.

To configure the schedulers, follow these steps:

1. Create a scheduler template as shown in Example 11-21.

Example 11-21 Scheduler template definition

```
class-of-service {
    schedulers name {
        priority strict-high|high|medium-high|medium-low|low;
            transmit-rate (rate | percent percentage | remainder) [exact];
    }
}
```

2. Define a scheduler map that associates each scheduler with a forwarding class as shown in Example 11-22.

Example 11-22 Scheduler map definition

```
edit class-of-service {
   scheduler-maps {
                map-name {
                        forwarding-class class-name scheduler scheduler-name;
                }
   }
}
```

3. Apply the scheduler map to the interface as shown in Example 11-23. Generally, you can associate schedulers with physical interfaces only. (For some interfaces, you can also associate schedulers with the logical interface.)

Example 11-23 Applying a scheduler map to an interface

```
class-of-service {
    interfaces interface-name {
        scheduler-map scheduler-map-name;
    }
}
```

11.10.3 Example of a scheduler configuration

Example 11-24 begins the configuration by defining the scheduler named TEST-SCHEDULER and assigning 20% of the bandwidth. The `exact` keyword limits the transmit rate, even though remaining bandwidth is in the other queues. In addition, low priority is configured. Then a scheduler map is defined, and the previous scheduler is associated with the best-effort forwarding class. Finally, the scheduler map is applied to the ge-0/0/0 interface.

Example 11-24 Scheduler configuration

```
{master}[edit]
ibm@J11M-re0# set class-of-service schedulers TEST-SCHEDULER transmit-rate percent
20 exact

{master}[edit]
ibm@J11M-re0# set class-of-service schedulers TEST-SCHEDULER priority low

{master}[edit]
ibm@J11M-re0# show class-of-service schedulers
TEST-SCHEDULER {
    transmit-rate percent 20 exact;
    priority low;
}

{master}[edit]
ibm@J11M-re0# set class-of-service scheduler-maps TEST-SCHEDULER-MAP
forwarding-class best-effort scheduler TEST-SCHEDULER

{master}[edit]
ibm@J11M-re0# set class-of-service interfaces ge-0/0/0 scheduler-map
TEST-SCHEDULER-MAP

{master}[edit]
ibm@J11M-re0# show class-of-service interfaces
ge-0/0/0 {
    scheduler-map TEST-SCHEDULER-MAP;
}

{master}[edit]
ibm@J11M-re0# commit

{master}[edit]
ibm@J11M-re0# run show class-of-service interface ge-0/0/0
Physical interface: ge-0/0/0, Index: 132
Queues supported: 8, Queues in use: 4
  Scheduler map: TEST-SCHEDULER-MAP, Index: 17100

  Logical interface: ge-0/0/0.0, Index: 83
    Object                  Name                    Type                    Index
    Classifier              ipprec-compatibility    ip                         13

{master}[edit]
ibm@J11M-re0# run show class-of-service scheduler-map TEST-SCHEDULER-MAP
Scheduler map: TEST-SCHEDULER-MAP, Index: 17100

  Scheduler: TEST-SCHEDULER, Forwarding class: best-effort, Index: 49130
```

```
    Transmit rate: 20 percent, Rate Limit: exact, Buffer size: remainder,
Priority: low
    Excess Priority: unspecified
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    <default-drop-profile>
      Medium low      any             1    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>
```

11.11 RED profiles

Junos software implements the RED algorithm, which was designed to prevent TCP synchronization when links experience congestion. When data networks become congested and drop packets, TCP sessions that suffer packet loss reduce their window size to avoid congestion. Indiscriminate packet drops (statistically) signal the transmitting endpoints to slow their transmission rates, causing most of the senders to exponentially decrease their transmission rates simultaneously. This phenomenon is known as *TCP synchronization*, and it leads to bandwidth fluctuations on congested links. When synchronization occurs, senders reduce their transmission rates simultaneously and slowly increase them again until links become congested. This process then repeats itself.

The RED algorithm solves this problem by randomly dropping packets as queues become full. The drop probability can be configured as a function of queue size at any given time. Therefore, the more congestion there is, the more aggressive the drop profile is. Randomly dropping traffic before an interface becomes congested signals end hosts to slow down, preventing an overloaded queue.

You configure RED by using a drop profile, which is a mechanism that defines parameters that allow packets to be dropped from the network. Drop profiles define the meanings of the loss priorities.

Configuring drop profiles entails setting two important values: the *queue fullness* and the *drop probability*. The queue fullness represents a percentage of the memory that is used to store packets in relation to the total amount that is allocated for that specific queue. Similarly, the drop probability is a percentage value that correlates to the likelihood that an individual packet is dropped from the network.

11.11.1 Configuration

You can configure loss profiles in the following ways:

- Staircase type profile

  You can specify a set of fill levels with the associated drop probabilities. The drop probability is assumed to be constant between fill levels.

- Interpolated profile

  You can create piecewise linear functions to specify the drop probability.

Loss profiles are configured under the [class-of-service drop-profiles] hierarchy as shown in Example 11-25.

Example 11-25 RED drop profile definition

```
[edit class-of-service]
   drop-profiles {
                profile-name {
                        fill-level percentage drop-probability percentage;
                             interpolate {
                                     drop-probability [ values ];
                                     fill-level [ values ];
                             }
                }
   }
```

After you configure the drop profiles, you must apply them to a scheduler, as shown in Example 11-26. For more information about schedulers, see 11.10, “Schedulers” on page 352.

Example 11-26 Applying a drop profile to a scheduler

```
class-of-service {
 schedulers {
        drop-profile-name {
            drop-profile-map loss-priority priority protocol any drop-profile
name;
        }
    }
}
```

11.11.2 Example

Example 11-27 shows how to generate the staircase profile specified in Figure 11-2 on page 358.

Example 11-27 Staircase RED profile configuration

```
{master}[edit]
ibm@J11M-re0# set class-of-service drop-profiles STAIR-TEST fill-level 70
drop-probability 10

{master}[edit]
ibm@J11M-re0# set class-of-service drop-profiles STAIR-TEST fill-level 80
drop-probability 20

{master}[edit]
ibm@J11M-re0# set class-of-service drop-profiles STAIR-TEST fill-level 90
drop-probability 25

ibm@J11M-re0# show class-of-service drop-profiles
STAIR-TEST {
    fill-level 70 drop-probability 10;
    fill-level 80 drop-probability 20;
    fill-level 90 drop-probability 25;
```

```
{master}[edit]
ibm@J11M-re0# commit

{master}[edit]
ibm@J11M-re0# run show class-of-service drop-profile STAIR-TEST
Drop profile: STAIR-TEST, Type: discrete, Index: 52959
  Fill level    Drop probability
          70                 10
          80                 20
          90                 25
```

Figure 11-2 Staircase RED profile

Example 11-28 shows how to generate the interpolated profile specified in Figure 11-3 on page 360.

Example 11-28 Interpolated RED profile configuration

```
{master}[edit]
ibm@J11M-re0# set class-of-service drop-profiles INTERPOLATE-TEST interpolate
fill-level fill-level [60 70 80 90 ] drop-probability [0 15 35 ]

{master}[edit]
ibm@J11M-re0# show class-of-service drop-profiles
INTERPOLATE-TEST {
    interpolate {
        fill-level [ 60 70 80 90 ];
        drop-probability [ 0 5 15 35 ];
    }
}}

{master}[edit]
ibm@J11M-re0# commit

{master}[edit]
ibm@J11M-re0# run show class-of-service drop-profile
Drop profile: <default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: INTERPOLATE-TEST, Type: interpolated, Index: 7093
  Fill level    Drop probability
           0                   0
```

1	0
2	0
4	0
5	0
6	0
8	0
10	0
12	0
14	0
15	0
16	0
18	0
20	0
22	0
24	0
25	0
26	0
28	0
30	0
32	0
34	0
35	0
36	0
38	0
40	0
42	0
44	0
45	0
46	0
48	0
49	0
51	0
52	0
54	0
55	0
56	0
58	0
60	0
62	1
64	2
65	2
66	3
68	4
70	5
72	7
74	9
75	10
76	11
78	13
80	15
82	19
84	23
85	25
86	27
88	31

90	35
92	48
94	61
95	67
96	74
98	87
99	93
100	100

Figure 11-3 Interpolated RED profile

11.12 Tail drop profiles

Tail drop profiles are a congestion management mechanism that allows a switch to drop arriving packets when queue buffers become full or begin to overflow. When you configure tail drop profiles, you set the value for queue fullness. The queue fullness represents a percentage of the memory that is used to store packets in relation to the total amount that has been allocated for that specific queue.

The queue fullness defines the delay-buffer bandwidth, which provides packet buffer space to absorb burst traffic up to the specified duration of delay. After the specified delay buffer becomes full, packets with 100% drop probability are dropped from the tail of the buffer.

By default, if you do not configure any drop profile, when the fill level is 0%, the drop probability is 0%. When the fill level is 100%, the drop probability is 100%.

11.12.1 Configuration

You configure tail-drop profiles under the `[class-of-service]` hierarchy as shown in Example 11-29. Then you apply the profile to a scheduler as shown in Example 11-26 on page 357 and explain in 11.10, “Schedulers” on page 352.

Example 11-29 Tail-drop profile definition

```
class-of-service {
    drop-profiles {
        name {
            fill-level level;
        }
    }
}
```

11.12.2 Example

Example 11-30 shows how to create a tail drop profile named TAIL with a 25% fill level.

Example 11-30 Creating a tail drop profile named TAIL

```
master:0}[edit]
ibm@J48E-VC# set class-of-service drop-profiles TAIL fill-level 25

{master:0}[edit]
ibm@J48E-VC# show class-of-service drop-profiles TAIL
fill-level 25;
```

11.13 Case study

This section shows the configuration of the following class-of-service features in the scenario shown in Figure 11-4:

- MF classification

  HTTP traffic from J02M is classified at the ingress in J11M and assigned to the EF forwarding class.

- Policing

  The rest of the traffic from J02M is assigned to the BE forwarding class with high loss priority and is policed to 1 Mbps.

- Rewrite marking

  Rewrite DSCP default marking is configured at the egress on J11M. The goal is to mark HTTP traffic classified in the previous task with the DSCP value of 101110. (To verify the configuration, a BA classifier is required in J48E-VC.)

- Scheduler

  Scheduler maps are configured on J11M to limit BE traffic to 5% of the interface bandwidth and to guarantee the EF class 20% of the interface bandwidth.

Purpose: The purpose of this section is to clarify the concepts introduced in this chapter. Carefully plan any class-of-service implementation according with the requirements of each design.

Figure 11-4 Study case diagram

Multifield classification and policing

Example 11-31 shows the firewall and policers settings on J11M.

Example 11-31 Firewall and policers settings

```
{master}[edit]
ibm@J11M-re0# set firewall filter MF-CLASSIFICATION term HTTP from protocol tcp
ibm@J11M-re0# set firewall filter MF-CLASSIFICATION term HTTP from port 80
ibm@J11M-re0# set firewall filter MF-CLASSIFICATION term HTTP then
forwarding-class expedited-forwarding
ibm@J11M-re0# set firewall filter MF-CLASSIFICATION term OTHER then policer
OTHER-POLICER

{master}[edit]
ibm@J11M-re0# set firewall policer OTHER-POLICER if-exceeding bandwidth-limit 1m
ibm@J11M-re0# set firewall policer OTHER-POLICER if-exceeding burst-size-limit 15k
ibm@J11M-re0# set firewall policer OTHER-POLICER then loss-priority high

ibm@J11M-re0# show firewall filter MF-CLASSIFICATION
term HTTP {
    from {
        protocol tcp;
        port 80;
    }
    then forwarding-class expedited-forwarding;
}
term OTHER {
    then policer OTHER-POLICER;
}

{master}[edit]
ibm@J11M-re0# show firewall policer OTHER-POLICER
if-exceeding {
    bandwidth-limit 1m;
    burst-size-limit 15k;
}
then loss-priority high;

{master}[edit]
ibm@J11M-re0# commit
```

To verify the configuration, check the counters on the output interface of J11M as shown in the following outputs:

1. Clear the counters on the interface:

```
{master}[edit]
ibm@J11M-re0# run clear interfaces statistics xe-1/1/0
```

2. Verify the queue counters:

```
ibm@J11M-re0# run show interfaces xe-1/1/0 detail | find "Queue counters"
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort                    0                    0                    0
    1 expedited-fo                   0                    0                    0
    2 assured-forw                   0                    0                    0
    3 network-cont                   0                    0                    0
```

3. Use Telnet from J02 to simulate TCP traffic on port 80:

```
[edit]
ibm@J02M# run telnet 192.168.200.1 port 80
Trying 192.168.200.1...
telnet: connect to address 192.168.200.1: Connection refused
telnet: Unable to connect to remote host
```

4. Verify the queue counters:

```
{master}[edit]
ibm@J11M-re0# run show interfaces xe-1/1/0 detail | find "Queue counters"
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort                    0                    0                    0
    1 expedited-fo                   1                    1                    0
    2 assured-forw                   0                    0                    0
    3 network-cont                   0                    0                    0
```

As expected, you can see that one packet was placed in the Expedited Forwarding queue.

5. Ping from J02M to verify policer configuration:

```
[edit]
ibm@J02M# run ping 192.168.200.1 size 1500 rapid count 100
PING 192.168.200.1 (192.168.200.1): 1500 data bytes
!!!
```

Verify the configuration and that the table was applied to the specified interface:

1. To verify the configuration, check the default DSCP rewrite table:

```
{master}[edit]
ibm@J11M-re0# run show class-of-service rewrite-rule type dscp
Rewrite rule: dscp-default, Code point type: dscp, Index: 31
  Forwarding class                    Loss priority       Code point
  best-effort                         low                 000000
  best-effort                         high                000000
  expedited-forwarding                low                 101110
  expedited-forwarding                high                101110
  assured-forwarding                  low                 001010
  assured-forwarding                  high                001100
  network-control                     low                 110000
  network-control                     high                111000
```

2. Verify that the table was applied to the specified interface:

```
{master}[edit]
ibm@J11M-re0# run show class-of-service interface xe-1/1/0
Physical interface: xe-1/1/0, Index: 227
Queues supported: 8, Queues in use: 4
  Scheduler map: <default>, Index: 2

  Logical interface: xe-1/1/0.0, Index: 74
    Object                  Name                    Type                      Index
    Rewrite                 dscp-default            dscp                         31
    Classifier              ipprec-compatibility    ip                           13
```

You can also generate traffic to the end host and check the interface counters at the egress port on J48E-VC:

1. Verify counters on J48E-VC egress port before generating traffic:

```
ibm@J48E-VC> show interfaces ge-1/0/46 detail | find "Queue counters"
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort                    0                    4                    0
    1 assured-forw                   0                    0                    0
    5 expedited-fo                   0                    0                    0
    7 network-cont                   0                    1                    0
  Active alarms  : None
  Active defects : None
```

2. Use Telnet from J02M to the end station on port 80:

```
ibm@J02M> telnet 10.100.100.2 port 80
Trying 10.100.100.2...
```

3. Again, verify counters on J48E-VC egress port:

```
ibm@J48E-VC> show interfaces ge-1/0/46 detail | find "Queue counters"
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort                    0                  850                    0
    1 assured-forw                   0                    0                    0
    5 expedited-fo                   0                    0                    0
    7 network-cont                   0                 1038                    0
```

The EF queue counters remained in 0 because the correct BA classifier must be applied at the ingress interface on J48E-VC. (By default, the ieee-802.1p classifier is applied to the ports on the J48E.)

4. Configure a BA classifier to assign DSCP 101110 to the EF queue and apply it to the ingress interface:

```
{master:1}[edit]
ibm@J48E-VC# set class-of-service classifiers dscp BA-CLASSIFICATION import default
forwarding-class expedited-forwarding loss-priority low code-points 101110
ibm@J48E-VC# set class-of-service interfaces xe-1/1/0 unit 0 classifiers dscp
BA-CLASSIFICATION
ibm@J48E-VC# commit
```

5. Use Telnet again from J02M to the end station on port 80:

```
ibm@J02M> telnet 10.100.100.2 port 80
Trying 10.100.100.2...
```

6. Verify the counters on the J48E-VC egress port:

```
ibm@J48E-VC> show interfaces ge-1/0/46 detail | find "Queue counters"
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 best-effort                    0                 1063                    0
    1 assured-forw                   0                    0                    0
    5 expedited-fo                   0                    3                    0
    7 network-cont                   0                  892                    0
```

As expected, TCP port 80 packets were assigned to the EF queue.

Scheduler

Example 11-33 shows how to configure the scheduler and scheduler map.

Example 11-33 Scheduler configuration

```
{master}[edit]
ibm@J11M-re0# set class-of-service schedulers BE transmit-rate percent 5 exact
ibm@J11M-re0# set class-of-service schedulers EF transmit-rate percent 20
ibm@J11M-re0# set class-of-service scheduler-maps SCHED-MAP forwarding-class
best-effort scheduler BE
ibm@J11M-re0# set class-of-service scheduler-maps SCHED-MAP forwarding-class
expedited-forwarding scheduler EF
ibm@J11M-re0# set class-of-service interfaces xe-1/1/0 scheduler-map SCHED-MAP
```

Verify the configuration of the scheduler map and that it is correctly applied to the interface:

1. Verify the configuration of the scheduler map:

```
ibm@J11M-re0# run show class-of-service scheduler-map SCHED-MAP
Scheduler map: SCHED-MAP, Index: 25389

  Scheduler: BE, Forwarding class: best-effort, Index: 2053
    Transmit rate: 5 percent, Rate Limit: exact, Buffer size: remainder,
Priority: low
    Excess Priority: unspecified
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    <default-drop-profile>
      Medium low      any             1    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>

  Scheduler: EF, Forwarding class: expedited-forwarding, Index: 2278
```

```
    Transmit rate: 20 percent, Rate Limit: none, Buffer size: remainder,
Priority: low
    Excess Priority: unspecified
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    <default-drop-profile>
      Medium low      any             1    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>
```

2. Verify if it is correctly applied to the interface:

```
ibm@J11M-re0# run show class-of-service interface xe-1/1/0
Physical interface: xe-1/1/0, Index: 227
Queues supported: 8, Queues in use: 4
  Scheduler map: SCHED-MAP, Index: 25389

  Logical interface: xe-1/1/0.0, Index: 74
    Object                  Name                    Type                    Index
    Rewrite                 dscp-default            dscp                       31
    Classifier              ipprec-compatibility    ip                         13
```

11.14 More information

See the following resources for more information:

- For class of service on IBM j-type m-series Ethernet routers, see *JUNOS Software Class of Service Configuration Guide*, GA32-0738:

  http://www-01.ibm.com/support/docview.wss?rs=1334&context=SGPMM5&dc=DA400&uid=isg3T7000161&loc=en_US&cs=utf-8&lang=en

- For class of service on IBM j-type e-series Ethernet switches, see *JUNOS Software for EX Series Ethernet Switches, Release 10.1: Class of Service* at the following address:

  http://www.juniper.net/techpubs/en_US/junos10.1/information-products/pathway-pages/ex-series/cos.html

- For the Differentiated Services Model, see RFC 2475:

  http://www.ietf.org/rfc/rfc2475.txt

- For a single-rate three-color marker, see RFC 2697:

  http://www.ietf.org/rfc/rfc2697.txt

- For two-rate color marker, see RFC 2698:

  http://www.ietf.org/rfc/rfc2698.txt

12

Security

This chapter provides information about the security features of IBM j-type Ethernet switches and routers. It includes the following topics:

- Access security
- Firewall filters
- Port security
- 802.1X authentication

© Copyright IBM Corp. 2011. All rights reserved.

12.1 Access security

This section explains the techniques of securing and controlling access to the IBM j-type Ethernet switches and routers.

12.1.1 Access overview

Junos software provides initial remote access security. By default, it disables remote access and allows only console access where you must physically connect to the console port for initial configuration. You can establish remote access to the switches and routers by using out-of-band management and inband management.

Out-of-band management

A dedicated management Ethernet port allows out-of-band management. Out-of-band management allows a direct connection to the Routing Engine and provides a complete separation of transit and management traffic. Transit traffic cannot traverse through this interface. Therefore, it ensures that congestion or failure in the transit network do not affect the management network of the switches and routers.

Inband management

Inband management allows connections to the switches and routers by using the same interfaces where both management and transit traffic flow. This approach is simple to implement and requires no dedicated management resources. However, it has several disadvantages:

- Management and transit traffic co-exit and mix together in the same network. The management of the switches and routers might become inaccessible or intermittently accessible when any attack occurs in the network.
- It might be prone to replay attacks or wiretapping because of the mixture of transit and management in a network.

Security measures are required to provide inband management. This chapter is written with the understanding that inband management is allowed for router and switch communication. It includes examples on securing and restricting the inband access.

12.1.2 Securing access to switches and routers

The following sections explain the methods that you can use to access a switch or router. They also explain how to implement user authentication.

Access method

For management access, you can communicate with a router or switch from a remote console in several ways. The most common methods that are used today are Telnet and Secure Shell (SSH) applications.

Telnet

Telnet, by default, does not encrypt any data sent over the connection between the administrator and router including the password. Anyone who has access to the network in which the router or switch reside can easily intercept the plain text packet that is passing by. They can also obtain the login ID and password information by using any common utilities such as **`tcpdump`** or Wireshark.

With this security-related shortcoming, Telnet has seen an obvious drop in usage especially on the Internet, with users favoring SSH.

Example 12-1 shows how to enable the Telnet application in the Junos software.

Example 12-1 Enabling the Telnet application

```
{master:0}[edit]
ibm@J48E-VC# set system services telnet
```

SSH

SSH provides secure encrypted communications over a network, which ensures the confidentiality and integrity of data. SSH uses public-key cryptography to authenticate the remote computer and allows the remote computer to authenticate the user. Therefore, SSH is suitable for inband management.

Example 12-2 shows how to enable the SSH service.

Example 12-2 Enabling the SSH service

```
{master:0}[edit]
ibm@J48E-VC# set system services ssh
```

Several optional parameters control access, for example, to the number of concurrent SSH sessions and the maximum number of SSH sessions that can be established in one minute as shown in Example 12-3. These parameters are useful for protecting the router and switch against SYN flood DoS attacks on SSH port (TCP port 22).

Example 12-3 Optional SSH parameters

```
{master:0}[edit]
ibm@IBM-J48E# edit system services ssh <--- Move in Junos configuration hierarchy

{master:0}[edit system services ssh]
ibm@IBM-J48E# set connection-limit 10 rate-limit 5
```

To restrict SSH access, you can disable root account from SSH access while only allowing other non-root accounts. By default, the router and switch accept connections for both version 1 and version 2 of SSH. In general, version 2 is considered more secure than version 1 because of its security and feature improvement over version 1. Use version 2 whenever possible.

Example 12-4 shows the restriction on root account login and the version 2 setting.

Example 12-4 SSH root account and version setting

```
{master:0}[edit system services ssh]
ibm@J48E-VC# set root-login deny

{master:0}[edit system services ssh]
ibm@J48E-VC# set protocol-version v2
```

User creation and authentication

Junos software, by default, creates only a root account in the initial configuration. When you first log in to the router or switch, you must set the password for the root account. Otherwise, you are unable to proceed with the other configuration.

For every remote access to the router or switch using Telnet or SSH, you are prompted for a login ID and password. Only the valid user account and password grant you the access to the router or switch.

To create a user account and password in Junos software, you can use the commands as show in Example 12-5.

Example 12-5 Creating a user account

```
{master:0}[edit]
ibm@J48E-VC# edit system login <--- Move in Junos configuration hierarchy

{master:0}[edit system login]
ibm@J48E-VC# set user administrator authentication plain-text-password
New password:
Retype new password:

{master:0}[edit system login]
```

Creating a user account is insufficient. In addition, you must assign the user to a class. By default, three classes are defined by the Junos software as shown in Example 12-6.

Example 12-6 Assigning a class to a user account

```
{master:0}[edit system login]
ibm@J48E-VC# set user administrator class ?
Possible completions:
  <class>              Login class
  operator             permissions [ clear network reset trace view ]
  read-only            permissions [ view ]
  super-user           permissions [ all ]
  unauthorized         permissions [ none ]
{master:0}[edit system login]
ibm@J48E-VC# set user administrator class super-user

{master:0}[edit system login]
```

In Example 12-6, the *administrator account* is assigned a super-user class, which allows you to perform the configuration as a root account in configuration mode.

The user account created in Example 12-6 is considered a local user database. It is only stored in this router and can be used for local authentication.

Central authentication

The management of multiple switches and routers by many different personnel in a huge network can create a significant user account management problem. Creating each user account in every router and switch in the network can cause operational complexity with an inconsistent user account database and problematic user password management. To address this problem, you can use a central authentication service to simplify account management.

A centralized authentication system helps to reduce operational task by having only on one server for account creation and deletion. It also allows you to enable or disable an account to all switches and routers with a single change that does not require committing a configuration change to any router.

Junos software supports two protocols for central authentication of users that are Remote Authentication Dial In User Server (RADIUS) and Terminal Access Controller Access Control System Plus (TACACS+).

When selecting an authentication protocol, RADIUS is always selected for its multivendor IETF standard. RADIUS is more widely accepted than TACACS+ or other proprietary systems.

The implementation of both RADIUS and TACACS+ in Junos software is essentially the same as shown in Example 12-7 and Example 12-8.

Example 12-7 RADIUS implementation

```
{master:0}[edit]
ibm@J48E-VC#
set system authentication-order radius
set system radius-server 10.1.1.1 secret ibm123juniper timeout 5
set system radius-server 10.1.1.2 secret ibm123juniper timeout 5
```

Example 12-8 TACACS+ implementation

```
{master:0}[edit]
ibm@J48E-VC#
set system authentication-order tacplus
set system tacplus-server 10.2.2.1 secret ibm123juniper timeout 5
set system tacplus-server 10.2.2.2 secret ibm123juniper timeout 5
```

The **`authentication-order radius`** command is used to enable RADIUS authentication only. If the RADIUS server is unreachable, the password of the local user is used as shown in Example 12-9.

Example 12-9 Password of local user used when the RADIUS server is unreachable

```
J48E-VC (ttyp5)

login: ibm
Password:
Local password: <--- local user's password is used

--- JUNOS 10.1R1.8 built 2010-02-12 17:24:20 UTC
{master:0}
ibm@J48E-VC>
```

Two RADIUS servers are referred for authentication. If the first RADIUS server does not respond within the specified time, in this case 5 seconds, the router can try the next server. The router and RADIUS share a same secret key that ensures they are talking to the trusted peer.

12.2 Firewall filters

In 12.1, “Access security” on page 368, we discussed the use of authentication to secure the router and switch access. Although authentication helps to restrict the access, it does not completely prevent malicious or untrusted packets from reaching a particular process on the Routing Engine.

To enhance security on the router or switch and network, you can deploy packet filters to allow only certain traffic into the Routing Engine. On IBM j-type Ethernet switches and routers, these filters are called *firewall filters*. On Cisco devices, they are called an *access list*.

Firewall filters provide rules that define whether to permit or deny packets that are transiting on an interface on a router or switch from a source address to a destination address. You configure firewall filters to determine whether to permit or deny traffic before it enters or exits a port, VLAN, or Layer 3 (routed) interface to which the firewall filter is applied. An *ingress firewall filter* is applied to packets that are entering a network. An *egress firewall filter* is applied to packets that are exiting a network.

You can also configure firewall filters to subject packets to the following types of processing:

- Filtering
- Class-of-service (COS) marking (grouping similar types of traffic together and treating each type of traffic as a class with its own level of service priority)
- Traffic policing (controlling the maximum rate of traffic sent or received on an interface)

The firewall filters are stateless filters that do not store the state of a packet flow. If firewall filters are applied in the input and output directions of an interface, ensure that the rules are defined correctly for both ingress and egress traffic before committing the firewall filters.

12.2.1 Firewall filter types

The following firewall filter types are supported for IBM Ethernet Switch J48Es:

- Port (Layer 2) firewall filter (Ethernet-switching)

  Port firewall filters apply to Layer 2 switch ports. You can apply port firewall filters in both ingress and egress directions on a physical port.

- VLAN firewall filter (Ethernet-switching)

  VLAN firewall filters provide access control for packets that enter a VLAN, are bridged within a VLAN, and leave a VLAN. You can apply VLAN firewall filters in both ingress and egress directions on a VLAN. VLAN firewall filters are applied to all packets that are forwarded to or forwarded from the VLAN.

- Router (Layer 3) firewall filter (inet)

  You can apply a router firewall filter in both ingress and egress directions on Layer 3 (routed) interfaces and Routed VLAN Interfaces (RVI). You can also apply a router firewall filter in ingress direction on the loopback interface.

Loopback interfaces: Firewall filters that are configured on loopback interfaces are applied to packets transiting network interfaces only. They are not applied to packets that transit the management interface (me0) in IBM Ethernet switches or (fxp0) in IBM Ethernet routers.

IBM Ethernet routers support more types of firewall filters. The following protocols families are supported:

- IPV4 (inet)
- IPV6 (inet6)
- MPLS (mpls)
- VPLS (vpls)
- Circuit cross-connects (ccc)

- Bridge (bridge)
- Protocol-independent (any)

This chapter only provides information about the Layer 3 firewall filter, which is the same for both IBM Ethernet Switches and Routers. For more details about advanced firewall filter implementation, see *JUNOS Software Policy Framework Configuration Guide*, GA32-0704.

12.2.2 Firewall filter components

In a firewall filter, you first define the family address type, for example, `inet` for a Layer 3 firewall filter or `ethernet-switching` for a Layer 2 or VLAN firewall filter. You then define one or more terms that specify the filtering criteria and the action to take if a match occurs.

Each term can include the following components:

- Match conditions

  Match conditions are the fields or values that a packet must contain. You can define various match conditions, including the following conditions:

  - IP source address field
  - IP destination address field
  - Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) source port field
  - IP Protocol field
  - Internet Control Message Protocol (ICMP) packet type
  - TCP flags
  - interfaces
- Action

  Specify the action to take if a packet matches the match condition. Possible actions are to accept or discard the packet or to send the packet to a specific virtual routing interface. In addition, a packet can be counted to collect statistical information. If no action is specified for a term, the default action is to accept the packet.

12.2.3 Firewall filter processing

The order of the terms within a firewall filter configuration is important. Packets are tested against each term in the order in which the terms are listed in the firewall filter configuration. When a firewall filter contains multiple terms, the switch takes a top-down approach and compares a packet against the first term in the firewall filter.

If the packet matches the first term, the switch executes the action that is defined by that term to either permit or deny the packet. No other terms are evaluated. If the switch does not find a match between the packet and first term, it compares the packet to the next term in the firewall filter by using the same match process.

If no match occurs between the packet and the second term, the switch continues to compare the packet to each successive term defined in the firewall filter until a match is found. If a packet does not match any terms in a firewall filter, the default action is to discard the packet.

12.2.4 Examples

This section provides two examples that demonstrate the use of firewall filters:

- Securing a Routing Engine by using a loopback filter
- Transit traffic filter

Securing a Routing Engine by using a loopback filter

The goal of this section is to secure traffic that is destined for the routers or switches. The following protocols are to be filtered:

- Management services (SSH, Domain Name System (DNS), Network Time Protocol (NTP) and SNMP (Simple Network Management Protocol))
- Routing protocol (Open Shortest Path First (OSPF))
- Diagnostic and troubleshooting protocols (Internet Control Message Protocol (ICMP) and traceroute)

First, a filter `protect-re` is created with the first term named `allow-ssh` to allow SSH traffic from trusted IP addresses (10.10.10.0/24) as shown in Example 12-10.

Example 12-10 Allowing SSH traffic

```
{master:0}[edit]
ibm@J48E-VC# edit firewall family inet filter protect-re <--- Move in Junos
configuration hierarchy

{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC#
set term allow-ssh from source-address 10.10.10.0/24
set term allow-ssh from protocol tcp
set term allow-ssh from destination-port ssh
set term allow-ssh then accept

{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC# show
term allow-ssh {
    from {
        source-address {
            10.10.10.0/24;
        }
        protocol tcp;
        destination-port ssh;
    }
    then accept;
}
```

Then you create a term to allow DNS response traffic from a DNS server (208.67.222.222) as shown in Example 12-11.

Example 12-11 Allowing DNS traffic

```
{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC#
set term allow-dns from source-address 208.67.222.222/32
set term allow-dns from source-address 53.0.0.0/32
set term allow-dns from protocol udp
```

```
set term allow-dns then accept

ibm@J48E-VC# show
term allow-dns {
    from {
        source-address {
            208.67.222.222/32;
            53.0.0.0/32;
        }
        protocol udp;
    }
    then accept;
}
```

For NTP and SNMP traffic, two terms are created as shown in Example 12-12.

Example 12-12 Allowing NTP and SNMP traffic

```
{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC#
set term allow-ntp from source-address 10.10.2.20/32
set term allow-ntp from protocol udp
set term allow-ntp from source-port 123
set term allow-ntp then accept

set term allow-snmp from source-address 10.10.2.30/32
set term allow-snmp from protocol udp
set term allow-snmp from source-port snmp
set term allow-snmp from source-port snmptrap
set term allow-snmp then accept


{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC# show
term allow-ntp {
    from {
        source-address {
            10.10.2.20/32;
        }
        protocol udp;
        source-port 123;
    }
    then accept;
}
term allow-snmp {
    from {
        source-address {
            10.10.2.30/32;
        }
        protocol udp;
        source-port [ snmp snmptrap ];
    }
    then accept;
}
```

At this stage, you have allowed the necessary management traffic to the Routing Engine. Next, you create a term to allow the OSPF protocol packet.

Example 12-13 shows that the OSPF packet from any peer is allowed. You can specify certain peers to be allowed in this term. However, you must maintain the IP address of the peer whenever adding a new peer or deleting a peer. Alternately, you can secure OSPF peering by setting OSPF authentication in the OSPF configuration.

Example 12-13 Allowing OSPF traffic

```
{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC#
set term allow-ospf from protocol ospf
set term allow-ospf then accept

{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC# show
term allow-ospf {
    from {
        protocol ospf;
    }
    then accept;
}
```

Switches and routers are always a default gateway of hosts or servers. Thus, allowing ICMP and traceroute protocols makes network troubleshooting much easier. Yet certain protection must be in place against an attack such as an ICMP flood. Example 12-14 shows the ICMP with certain types allowed. ICMP traffic is also limited with a bandwidth policer in which ICMP traffic is discarded when the bandwidth exceeds 1m and 15k of burst limit. Unlike any Windows client that uses ICMP for traceroute, Junos software implements traceroute using a user-defined protocol (UDP) similar to other UNIX platform.

Example 12-14 Allowing ICMP and traceroute traffic

```
{master:0}[edit firewall family inet filter protect-re]

ICMP
ibm@J48E-VC#
set term allow-icmp from protocol icmp
set term allow-icmp from icmp-type echo-request
set term allow-icmp from icmp-type echo-reply
set term allow-icmp from icmp-type unreachable
set term allow-icmp from icmp-type time-exceeded
set term allow-icmp then policer icmp-policer
set term allow-icmp then accept

{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC# show
term allow-icmp {
    from {
        protocol icmp;
        icmp-type [ echo-request echo-reply unreachable time-exceeded ];
    }
```

```
    then {
        policer icmp-policer;
        accept;
    }
}
```

Traceroute

```
ibm@J48E-VC#
set term allow-traceroute from protocol udp
set term allow-traceroute from destination-port 33434-33523
set term allow-traceroute then accept

{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC# show
term allow-traceroute {
    from {
        protocol udp;
        destination-port 33434-33523;
    }
    then accept;
}
```

Policer

```
{master:0}[edit]
ibm@J48E-VC# edit firewall policer icmp-policer <--- Move in the Junos
configuration hierarchy

{master:0}[edit firewall policer icmp-policer]
ibm@J48E-VC#
set if-exceeding bandwidth-limit 1m
set if-exceeding burst-size-limit 15k
set then discard

{master:0}[edit firewall policer icmp-policer]
ibm@J48E-VC# show
if-exceeding {
    bandwidth-limit 1m;
    burst-size-limit 15k;
}
then discard;
```

Optionally, you can use a term to count all denied traffic and to log to a syslog file as shown in Example 12-15.

Example 12-15 Counting and logging denied traffic

```
{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC#
set term denied-traffic then count denied-packet
set term denied-traffic then discard

{master:0}[edit firewall family inet filter protect-re]
ibm@J48E-VC# show
term denied-traffic {
    then {
        count denied-packet;
```

```
        discard;
    }
}
```

After the `protect-re` firewall filter is defined (as described in Example 12-10 on page 374 to Example 12-15 on page 377), you must apply the firewall filter to the loopback interface as an input filter as shown in Example 12-16.

Example 12-16 Applying a firewall filter to the loopback interface

```
{master}[edit]
ibm@J48E-VC# set interfaces lo0.0 family inet filter input protect-re
```

Transit traffic filter

This section provides information about the Layer 3 firewall filter that filters traffic between two networks as shown in Figure 12-1.

Figure 12-1 Applying the Layer 3 firewall filter

The example in this section has the following goals:

- TCP port 25 of the email server can be accessed by Host 1 and Host 2.
- UDP port 514 of the syslog server can be accessed by only Host 3.
- Ping is allowed in both networks.

First, a firewall filter called `protect-server` is created. You define the first term to allow access to the email server from Host 1 and Host 2 as shown in Example 12-17.

Example 12-17 Allowing access to the email server

```
{master:0}[edit]
ibm@J48E-VC# edit firewall family inet filter protect-server <--- Move in the
Junos configuration hierarchy

{master:0}[edit firewall family inet filter protect-server]
```

```
ibm@J48E-VC#
set term email-server from source-address 10.8.8.101/32
set term email-server from source-address 10.8.8.102/32
set term email-server from destination-address 10.10.10.20/32
set term email-server from protocol tcp
set term email-server from destination-port 25
set term email-server then accept

{master:0}[edit firewall family inet filter protect-server]
ibm@J48E-VC# show
term email-server {
    from {
        source-address {
            10.8.8.101/32;
            10.8.8.102/32;
        }
        destination-address {
            10.10.10.20/32;
        }
        protocol tcp;
        destination-port 25;
    }
    then accept;
}
```

Then, create second term to allow access to the syslog server from only Host 3 as shown in Example 12-18.

Example 12-18 Allowing access to the syslog server

```
{master:0}[edit firewall family inet filter protect-server]
ibm@J48E-VC#
set term syslog-server from source-address 10.8.8.103/32
set term syslog-server from destination-address 10.10.10.30/32
set term syslog-server from protocol udp
set term syslog-server from destination-port 514
set term syslog-server then accept

{master:0}[edit firewall family inet filter protect-server]
ibm@J48E-VC# show
term syslog-server {
    from {
        source-address {
            10.8.8.103/32;
        }
        destination-address {
            10.10.10.30/32;
        }
        protocol udp;
        destination-port 514;
    }
    then accept;
}
```

To allow ping packets in both network, define a term as shown in Example 12-19.

Example 12-19 Allowing a ping packet

```
{master:0}[edit firewall family inet filter protect-server]
ibm@J48E-VC#
set term ping from destination-address 10.10.10.0/24
set term ping from protocol icmp
set term ping then accept

{master:0}[edit firewall family inet filter protect-server]
ibm@J48E-VC# show
term ping {
    from {
        destination-address {
            10.10.10.0/24;
        }
        protocol icmp;
    }
    then accept;
}
```

A final deny term is added at the end of the filter. Although this action is the default behavior for a filter, it is explicitly configured here to count the denied traffic. Example 12-20 shows the `denied-traffic` term for counting a denied packet and the output to view the number of bytes and packets of the `denied-packet` counter.

Example 12-20 Count denied traffic

```
{master:0}[edit firewall family inet filter protect-server]
ibm@J48E-VC#
set term denied-traffic then count denied-packet
set term denied-traffic then discard

{master:0}[edit firewall family inet filter protect-server]
ibm@J48E-VC# show
term denied-traffic {
    then {
        count denied-packet;
        discard;
    }
}


ibm@J48E-VC> show firewall

Filter: protect-server
Counters:
Name                                                Bytes              Packets
denied-packet                                           0                    0
```

The firewall filter `protect-server` must apply to the interface ge-1/0/1 as an input filter as shown in Example 12-21.

Example 12-21 Applying a firewall filter to interface ge-1/0/1

```
{master:0}[edit]
ibm@J48E-VC# set interfaces ge-1/0/1.0 family inet filter input protect-server
```

12.3 Port security

Ethernet LANs are vulnerable to attacks such as address spoofing (forging) and Layer 2 denial-of-service (DoS) attacks on network devices. Port security features help protect the access ports on your switch against the loss of information and productivity that can result from such attacks.

You can turn on port security features to obtain the most robust port security level. Basic port security features are enabled in the default configuration of the switch. You can configure additional features with minimal configuration steps.

Port security in Junos software: As of Junos version 10.1R1.8, port security features are only supported on IBM Ethernet Switch J48E. IBM Ethernet Switch J08E and J16E and IBM Ethernet Routers (J02M, J06M, and J11M) are not supported yet.

12.3.1 Features of port security

IBM Ethernet Switches J48s have the following port security features:

- Dynamic Host Configuration Protocol (DHCP) snooping

  This feature filters and blocks ingress DHCP server messages on untrusted ports. It also builds and maintains an IP address or Media Access Control (MAC) address binding database (called the *DHCP snooping database*). You enable this feature on VLANs.

- Dynamic ARP inspection (DAI)

  This feature prevents Address Resolution Protocol (ARP) spoofing attacks. ARP requests and replies are compared against entries in the DHCP snooping database. Filtering decisions are made based on the results of those comparisons. You enable this feature on VLANs.

- MAC limiting

  This feature protects against flooding of the Ethernet switching table (also known as the *MAC forwarding table* or *Layer 2 forwarding table*). You enable this feature on access interfaces (ports).

- MAC move limiting

  This feature detects MAC movement and MAC spoofing on access ports. You enable this feature on VLANs.

- Trusted DHCP server

  With a DHCP server on a trusted port, this feature protects against rogue DHCP servers sending leases. You enable this feature on interfaces (ports). By default, access ports are untrusted, and trunk ports are trusted. *Access ports* are the switch ports that connect to Ethernet endpoints, such as user PCs and mobile computers, servers, and printers. *Trunk ports* are the switch ports that connect to other Ethernet switches or to routers.

- IP source guard

  This feature mitigates the effects of IP address spoofing attacks on the Ethernet LAN. You enable this feature on VLANs. With IP source guard enabled, the source IP address in the packet sent from an untrusted access interface is validated against the source MAC address in the DHCP snooping database. The packet is allowed for further processing if the source IP address to source MAC address binding is valid. If the binding is invalid, the packet is discarded.

- DHCP option 82

  This feature is also known as the *DHCP relay agent information option*. It helps protect the EX Series switch against attacks such as the spoofing of IP addresses and MAC addresses and DHCP IP address starvation. Option 82 provides information about the network location of a DHCP client. The DHCP server uses this information to implement IP addresses or other parameters for the client.

12.3.2 MAC limiting and MAC move limiting

MAC limiting protects Ethernet switches from the attacks that use MAC addresses such as MAC flooding and MAC spoofing. Both attacks are harmful and can cause DoS to users and systems.

MAC limiting can be configured by using either of the following methods:

- Specify the maximum number of MAC addresses that can be learned on a single Layer 2 access port. This method prevents MAC flooding. When the switch reaches the configured maximum number of MAC addresses, every new MAC address learned on that access port is subjected to the configured action. If no action is configured, the default drop action occurs.
- Define the allowed MAC addresses on an access port. This method prevents MAC spoofing. Any MAC address that is not listed is not permitted to the network.

When a MAC address limit is exceed, the switch can perform one of the following actions:

Drop Drop the packet and generate an alarm, an SNMP trap, or a system log entry. This action is the default action.

Log Do not drop the packet, but generate an alarm, an SNMP trap, or a system log entry.

None No action.

Shutdown Disable the interface and generate an alarm. If you configured the switch with the **`port-error-disable`** statement, the disabled interface recovers automatically upon expiration of the specified disable timeout. If you have not configured the switch for auto recovery from port error disabled conditions, you can start the disabled interfaces by running the **`clear ethernet-switching port-error`** command.

Example: Configuring MAC limiting

This example, which is shown in Example 12-22 on page 383, has the following goals:

- Set the MAC limit of the ge-0/0/0 interface to 2. When the MAC address limit is exceeded, drop any new MAC address.
- Configure two static MAC addresses on the ge-0/0/1 interface: `00:1a:64:a0:ed:ae` and `00:1c:25:a0:20:42`.

Example 12-22 Configuring MAC limiting

```
{master:0}[edit]
ibm@J48E-VC# edit ethernet-switching-options <--- move in Junos configuration
hierarchy

{master:0}[edit ethernet-switching-options]
ibm@J48E-VC# edit secure-access-port <--- move in Junos configuration hierarchy

{master:0}[edit ethernet-switching-options secure-access-port]
set interface ge-0/0/0.0 mac-limit 2 action drop
set interface ge-0/0/1.0 allowed-mac 00:1a:64:a0:ed:ae
set interface ge-0/0/1.0 allowed-mac 00:1c:25:a0:20:42
```

After configuring the MAC limiting feature on the switch, you can use view the contents of the MAC table by using the **show ethernet-switching table** command. The MAC table shows the learned MAC address entries. Each MAC address entry includes the VLAN, type, age, and interface information as shown in Example 12-23.

Example 12-23 Viewing the MAC table

```
{master:0}
root@J48E-VC> show ethernet-switching table
Ethernet-switching table: 5 entries, 3 learned
  VLAN              MAC address       Type         Age Interfaces
  default           *                 Flood          - All-members
  default           00:1d:b5:c2:4f:c0 Learn          0 xe-0/1/0.0
  employee          *                 Flood          - All-members
  employee          00:0c:29:fe:d0:20 Learn          0 ge-0/0/0.0
  employee          00:1c:25:a0:20:42 Learn          0 ge-0/0/0.0
```

When a MAC limiting violation occurs, you can use the **show log messages | match limit** command to view the MAC limiting violations. Example 12-24 shows two types of messages that are being logged. The first message indicates dropping the packet that is triggered by a drop action. The second message indicates shutting down the interface that is triggered by a shutdown action.

Example 12-24 Viewing the MAC limiting log

```
{master:0}
root@J48E-VC> show log messages | match limit
Feb 21 14:25:47  J48E-VC eswd[11933]: ESWD_MAC_LIMIT_DROP: MAC limit (2) exceeded
at ge-0/0/0.0: dropping the packet

Feb 21 14:28:33  J48E-VC eswd[11933]: ESWD_MAC_LIMIT_BLOCK: MAC limit (2) exceeded
at ge-0/0/0.0: shutting down the interface
```

The current blocking state of an interface can be viewed by using the **show ethernet-switching interfaces <*interface name*> detail** command. Example 12-25 shows that the ge-0/0/0 interface has exceeded the MAC limit and the port is disabled.

Example 12-25 Viewing the port state

```
root@J48E-VC> show ethernet-switching interfaces ge-0/0/0 detail
Interface: ge-0/0/0.0, Index: 122, State: down, Port mode: Access
Ether type for the interface: 0x8100
VLAN membership:
```

```
    employee, 802.1Q Tag: 100, untagged, msti-id: 0, MAC limit exceeded
Number of MACs learned on IFL: 0

root@J48E-VC> clear ethernet-switching port-error
```

You can start the disabled port by entering the **clear ethernet-switching port-error** command or configure an automatic recovery feature from the error condition after a specified time period. Example 12-26 shows the manual way to re-enable the port and the automatic way to recover the port from an error state after 10 seconds.

Example 12-26 Resetting the disabled port

```
{master:0}
root@J48E-VC> clear ethernet-switching port-error

{master:0}[edit ethernet-switching-options]
root@J48E-VC# set port-error-disable disable-timeout 10
```

In some cases, you might need to clear the MAC table entries. You can clear all the entries or only the entries that are associated with an interface as shown in Example 12-27.

Example 12-27 Clearing the MAC table entries

```
{master:0}
root@J48E-VC> clear ethernet-switching table

{master:0}
root@J48E-VC> clear ethernet-switching table interface ge-0/0/0
```

Example: Configuring MAC move limiting

MAC move limiting causes the switch to track the number of times that a MAC address can move to a new interface (port). It can help to prevent MAC spoofing. It can also detect and prevent loops.

Unlike MAC limiting that applies on a per-port setting, MAC move limiting applies on a per-VLAN basis. A MAC move involves at least two ports. Therefore, the need to control the aggregate number of MAC moves with the context of a specific VLAN.

Example 12-28 illustrates the setting of MAC move limiting on employee VLAN. The VLAN shuts down when the number of MAC moves exceeds three times in a second.

Example 12-28 Configuring MAC move limiting

```
{master:0}[edit]
ibm@J48E-VC# edit ethernet-switching-options <--- move in Junos configuration
hierarchy

{master:0}[edit ethernet-switching-options]
ibm@J48E-VC# edit secure-access-port <--- move in Junos configuration hierarchy

{master:0}[edit ethernet-switching-options secure-access-port]
root@J48E-VC# set vlan employee mac-move-limit 3 action shutdown
```

12.3.3 Port security features in a Layer 2 network

The following sections provide information about three port security features that are usually implemented together in Layer 2 network:

- DHCP snooping
- ARP inspection
- IP Source Guard

DHCP snooping

Figure 12-2 shows a typical operation of DHCP in which the following actions occur:

- The DHCP client generates a DHCPDISCOVER message and broadcasts the message that is flooded throughout a Layer 2 domain.
- The DHCP server residing in the same Layer 2 domain responds with a unicast DHCPOFFER message to the client with a proposed set of configuration parameters.
- When the DHCP client receives the first response, it broadcasts a DHCPREQUEST message back to the DHCP server indicating the parameters that it wants to use.
- The DHCP server sends a unicast DHCPACK message to the DHCP client to confirm the assignment of the related parameters and provides a lease duration.

Figure 12-2 DHCP client and server exchange

Because the DHCP request message is broadcasted to the network, the DHCP request message can potentially be seen by all devices on the subnet. It is vulnerable when a rogue DHCP server intentionally responds to the client and provides bogus network parameters to disrupt normal network operations and start a DoS attack.

With the DHCP snooping feature, it filters and blocks ingress DHCP server messages on untrusted ports. It builds and maintains an IP address or MAC address binding database through monitoring of DHCP exchanges. Therefore, it protects the switch and other resources in the network from malicious attacks.

Figure 12-3 illustrates the basic process of DHCP snooping, which entails the following actions:

1. The device sends a DHCPDISCOVER message to request the IP address or a DHCPREQUEST message to accept the IP address and lease.
2. A switch snoops a packet. It adds the IP-MAC placeholder binding to the database.
3. The switch forwards the DHCPDISCOVER or DHCPREQUEST message.
4. The server sends the DHCPOFFER message to offer an address, and either DHCPACK to assign an address or DHCPNAK to deny an address request.
5. The switch snoops the packets. If a placeholder exists, the switch replaces it with the IP-MAC binding upon receipt of the DHCPACK message.
6. The switch forwards the DHCPOFFER, DHCPACK, or DHCPNAK message.

Figure 12-3 The DHCP snooping process

ARP inspection

The ARP inspection feature prevents ARP spoofing attacks by comparing ARP requests and replies against entries stored in the DHCP snooping database. ARP packets that do not match values in the snooping database are filtered.

IP Source Guard

The IP Source Guard feature prevents the effects of IP address spoofing attacks by validating the source IP address in packets received on an untrusted access interface against the DHCP snooping database. When the source IP address and source MAC address are found to be valid, the packet is accepted. If a mismatch occurs in either the source IP or source MAC address, the packet is discarded.

This feature is useful in a case where a rogue host connects to the network and attempts to spoof the IP address of a DHCP server to act maliciously as a DHCP server to start a DoS or other attack.

Example

This example demonstrates how to configure DHCP snooping, ARP inspection, and IP Source Guard. It has the following goals:

- Do not accept DHCP responses on interfaces ge-0/0/1, ge-0/0/2, and ge-0/0/3.
- Configure the ge-0/0/0 interface to accept trusted DHCP packets.
- Enable DHCP snooping on employee VLAN to allow a valid DHCP response on a trusted port.
- Configure Dynamic ARP Inspection on employee VLAN to prevent ARP spoofing.
- Configure IP Source Guard on employee VLAN to prevent IP spoofing.

Figure 12-4 shows the DHCP clients, switch, and DHCP server in the network.

Figure 12-4 Network topology for port security

Configuring DHCP snooping, ARP inspection, and IP Source Guard are straightforward. These configurations have the following two levels:

- VLAN level setting for enabling DHCP snooping, ARP inspection, and IP Source Guard
- Port level setting for trusted and untrusted DHCP or ARP configuration

Example 12-29 shows the setting on employee VLAN for DHCP snooping, ARP inspection, and IP Source Guard.

Example 12-29 Configuring DHCP snooping, ARP inspection, and IP Source Guard

```
{master:0}[edit]
ibm@J48E-VC# edit ethernet-switching-options <--- Move in the Junos configuration
hierarchy

{master:0}[edit ethernet-switching-options]
ibm@J48E-VC# edit secure-access-port <--- Move in the Junos configuration
hierarchy

{master:0}[edit ethernet-switching-options secure-access-port]
set vlan employee examine-dhcp
set vlan employee arp-inspection
set vlan employee ip-source-guard

{master:0}[edit ethernet-switching-options secure-access-port]
ibm@J48E-VC# show
vlan employee {
    arp-inspection;
    examine-dhcp;
    ip-source-guard;
}
```

DHCP snooping is enabled for employee VLAN by using the **examine-dhcp** command. After DHCP snooping is enabled, you can turn on ARP inspection and IP Source Guard. Both ARP

inspection and IP Source Guard require a DHCP snooping database to work. Therefore, DHCP snooping is a prerequisite for ARP inspection and IP Source Guard. These features have no optional configuration, but enabling and disabling with a command.

Example 12-30 shows a port-level configuration.

Example 12-30 Configuring a trusted port for the DHCP server

```
{master:0}[edit ethernet-switching-options secure-access-port]
set interface ge-0/0/0 dhcp-trusted
set interface ge-0/0/1 no-dhcp-trusted
set interface ge-0/0/2 no-dhcp-trusted
set interface ge-0/0/3 no-dhcp-trusted

{master:0}[edit ethernet-switching-options secure-access-port]
ibm@J48E-VC# show
interface ge-0/0/0.0 {
    dhcp-trusted;
}
interface ge-0/0/1.0 {
    no-dhcp-trusted;
}
interface ge-0/0/2.0 {
    no-dhcp-trusted;
}
interface ge-0/0/3.0 {
    no-dhcp-trusted;
}
```

DHCP snooping with the examine-dhcp keyword: After DHCP snooping is enabled with the `examine-dhcp` keyword, all trunk ports are considered DHCP trusted and all access ports are considered untrusted.

In Example 12-30, you specifically configure the ge-0/0/0 interface to be a trusted DHCP port to allow a DHCP server packet from this port. Example 12-30 also shows the commands to configure the other access ports to be untrusted DHCP ports that prohibit DHCP server packets. Although the untrusted access port configuration is not needed in this case, it is only shown here to alter the defaults on a per-port basis.

With the DHCP snooping configured and the DHCP server connected to the switch, you now connect Host 1 and Host 2 to the switch. Host 1 and Host 2 are assigned an IP address respectively. The switch should have built a DHCP snooping database for these hosts. Example 12-31 shows the IP to MAC address information of Host 1 and Host 2 in the DHCP snooping database. Host 1 is assigned 10.1.1.205, and Host 2 is assigned 10.1.1.208.

Example 12-31 Viewing the DHCP snooping database

```
root@J48E-VC> show dhcp snooping binding
DHCP Snooping Information:
MAC address        IP address        Lease (seconds)  Type     VLAN    Interface
00:1A:64:A0:ED:AE  10.1.1.208                  86020  dynamic  employee ge-0/0/2.0
00:1C:25:A0:20:42  10.1.1.205                  86018  dynamic  employee ge-0/0/1.0
```

To view the results of dynamic ARP inspection, you enter the **show arp inspection statistics** command as shown in Example 12-32.

Example 12-32 Viewing the ARP inspection statistics

```
root@J48E-VC> show arp inspection statistics
Interface     Packets received     ARP inspection pass  ARP inspection failed
  ge-0/0/0                   3                       3                      0
  ge-0/0/1                   4                       4                      0
  ge-0/0/2                   3                       3                      0
  ge-0/0/3                   0                       0                      0
  ge-0/0/4                   0                       0                      0
```

The output of the ARP inspection results clearly indicates that ARP responses on the ge-0/0/1 and ge-0/0/2 interfaces are valid and match the content of the DHCP snooping database. The number of packets received on the port is the same as the number that passed ARP inspection.

To test ARP inspection with ARP spoofing, change the MAC address of Host 1 to the same MAC address of Host 2 and then use PING to test the DHCP server. Again, check the ARP inspection statistics. You can see that the ARP inspection has prevented ARP cache poisoning by blocking the flooding of the bogus ARP message in the VLAN as shown in Example 12-33. As expected, you can see that the ARP inspection has failed the sending packets.

Example 12-33 ARP inspection statistics after ARP poisoning

```
root@J48E-VC> show arp inspection statistics
Interface     Packets received     ARP inspection pass  ARP inspection failed
  ge-0/0/0                  71                      71                      0
  ge-0/0/1                  24                      23                      1
  ge-0/0/2                  16                       3                     13
```

You can view IP Source Guard information by using the **show ip-source-guard** command as shown in Example 12-34.

Example 12-34 Viewing the IP source guard information

```
root@J48E-VC> show ip-source-guard
IP source guard information:
Interface    Tag   IP Address       MAC Address         VLAN
ge-0/0/1.0   0     10.1.1.205       00:1C:25:A0:20:42   employee
ge-0/0/2.0   0     10.1.1.208       00:1A:64:A0:ED:AE   employee
```

Because the DHCP snooping database is referred by IP Source Guard, you might notice that the IP Source Guard Information is similar to DHCP snooping.

In this case, you can test the IP Source Guard feature by changing the IP address of Host 1 to the same IP address of Host 2, which is 10.1.1.208. Then, you perform a few PING tests to the DHCP server from Host 1 and Host 2. The result is that Host 2 can issue a PING to the DHCP server, and Host 1 fails to PING. This result indicates that IP Source Guard has prevented IP spoofing of Host 1 and ensured that Host 2 is operating as normal.

12.4 802.1X authentication

802.1X is an Institute of Electrical and Electronics Engineers (IEEE) standard used for port-based network access control and authentication. 802.1X does not replace other security technologies. It works with port security features, such as DHCP snooping, ARP inspection and MAC limiting, to guard against DoS attacks and spoofing.

The 802.1X feature in Junos software: As of Junos version 10.1R1.8, the 802.1X feature is only supported on IBM Ethernet Switch J48E. IBM Ethernet Switch J08E and J16E and IBM Ethernet Routers J02M, J06M, and J11M are not supported yet.

12.4.1 Overview of 802.1X authentication

An 802.1X authentication setup contains three basic components (illustrated in Figure 12-5):

Supplicant
The IEEE term for a host that requests to join the network. The host can be responsive or nonresponsive. A *responsive host* is one on which 802.1X is enabled and provides authentication credentials. Specifically, these credentials include a user name and password for Extensible Authentication Protocol (EAP)-message-digest algorithm 5 (MD5) or a user name and client certificates for EAP-TLS, EAP-TTLS, and EAP-PEAP. A *nonresponsive host* is one on which 802.1X is not enabled, but can be authenticated by using a MAC-based authentication method.

Authenticator Port Access Entity
The IEEE term for the authenticator. The IBM Ethernet Switch J48E is the authenticator. It controls access by blocking all traffic to and from supplicants until they are authenticated.

Authentication server
The authentication server contains the backend database that makes authentication decisions. It contains credential information for each supplicant that can connect to the network. The authenticator forwards credentials supplied by the supplicant to the authentication server. If the credentials match the credentials in the authentication server database, access is granted. If the credentials do not match, access is denied. The IBM Ethernet Switch J48Es support RADIUS authentication servers.

Figure 12-5 802.1X components

The 802.1X standard and authentication process

The 802.1X standard is based on Extensible Authentication Protocol (EAP), which is a universal authentication framework. EAP is not an authentication mechanism by itself. Instead, EAP provides common functions and a negotiation method to determine the

authentication mechanism (EAP method) that is used between the supplicant and the authentication server. EAP methods include IETF standards and proprietary standards.

The following EAP methods are supported on IBM Ethernet Switch J48Es:

- EAP-MD5
- EAP-TLS (Transport Layer Security)
- EAP-TTLS (Tunneled Transport Layer Security)
- EAP-PEAP (Protected EAP)

Figure 12-6 illustrates the authentication process.

Figure 12-6 802.1X authentication process

This process entails the following actions:

1. Either the client or the switch initiates the authentication. The client initiates authentication by sending an EAPOL-Start message. Alternatively, the switch initiates authentication when it receives the first data packet from the client. In this state, only 802.1X traffic is allowed. All other traffic is blocked at the data link layer.
2. The authenticator sends an EAP Request for the identity or credentials of the supplicant.
3. The supplicant replies with an EAP Response message.
4. The authenticator includes the credentials of the supplicant in a RADIUS Access Request message and then sends them to the authentication server.
5. The authentication server accepts or rejects the access request. When the access request is accepted, the authentication server sends a RADIUS Access Challenge to the authenticator.
6. The authenticator converts the RADIUS Access Challenge to an EAP Request for a challenge response and sends it back to the supplicants.
7. The supplicant returns the challenge response in the EAP format.
8. The authenticator forwards it to the RADIUS server.
9. If the response of the supplicant meets the expectations of the authentication server, a RADIUS Access Accept message is generated to the authenticator. Otherwise, an Access Reject message is sent. The RADIUS server can return a dynamically assigned VLAN or firewall filter in its Access Accept message through support for vendor-specific attributes.
10. The authenticator grants access when it receives an Access Accept message or block access for an Access Reject message.
11. The supplicant sends an EAP-Logoff upon disconnection from the network. In response, the authenticator returns the port to the unauthorized state, again blocking all non-EAP traffic as it patiently awaits its next supplicant.

Supplicant modes

The supplicant is authenticated in single mode, single-secure mode, or multiple mode:

single Authenticates only the first supplicant. All other supplicants who connect later to the port are allowed full access without any further authentication. They effectively “piggyback” on the authentication of the first supplicant.

single-secure Allows only one supplicant to connect to the port. No other supplicant is allowed to connect until the first supplicant logs out.

multiple Allows multiple supplicants to connect to the port. Each supplicant is authenticated individually.

802.1X features

The 802.1X has the following features on IBM Ethernet Switches:

- Guest VLANs provide limited access to a LAN, typically just to the Internet, for supplicants that fail 802.1X authentication.
- Dynamic VLANs enable a supplicant, after authentication, to be a member of a VLAN dynamically.
- Private VLANs enable configuration of 802.1X authentication on interfaces that are members of private VLANs (PVLANs).

- Dynamic changes to a user session allow the switch administrator to terminate an already authenticated session. This feature is based on support of the RADIUS Disconnect Message defined in RFC 3576.
- Support for voice over IP (VoIP) supports IP telephones. If the phone is 802.1X-enabled, it is authenticated similar to any other supplicant. If the phone is not 802.1X-enabled, but has another 802.1X-compatible device connected to its data port, that device is authenticated. Then VoIP traffic can flow to and from the phone, if the interface is configured in single mode and not in single-secure mode.

VoIP on PVLAN: Configuring a VoIP VLAN on PVLAN interfaces is not supported.

- RADIUS accounting sends accounting information to the RADIUS accounting server. Accounting information is sent to the server whenever a subscriber logs in or logs out and whenever a subscriber activates or de-activates a subscription.
- Vendor-specific attributes support the Juniper-Switching-Filter attribute on the RADIUS authentication server that can further define the acess of a supplicant during the 802.1X authentication process. Centrally configuring vendor-specific attributes on the authentication server removes the need to configure these same attributes in the form of firewall filters on every switch in the LAN to which the supplicant might connect to the LAN. This feature is based on RLI 4583, AAA RADIUS BRAS Vendor-specific Attribute (VSA) support.

12.4.2 Example: Configuring 802.1X authentication

This section explains how to configure and validate 802.1X authentication using both the EAP-MD5 and MAC-based methods. Figure 12-7 illustrates the topology of this example.

Figure 12-7 802.1X authentication topology

This example has the following goals:

- Configure the ge-0/0/1 interface for 802.1X authentication to support EAP-MD5. The RADIUS server should dynamically assign the trusted host a VLAN ID 100 that is employee VLAN.

- Configure the ge-0/0/2 interface to allow MAC-based authentication with a RADIUS server for the printer. The RADIUS server should dynamically assign the printer a VLAN ID 200 that is printer VLAN.
- Configure the ge-0/0/3 interface to allow visiting guests to connect to the network in a guest-access VLAN.
- Ensure that only one supplicant can be active. All other sources are blocked.
- Force the client to re-authenticate every 180 seconds.

The key assumptions are that the RADIUS server is configured properly to support 802.1X authentication, and the supplicant software is installed correctly in a trusted host. As the integrated components in this example, the key setting on the RADIUS server and trusted host are shown.

Radius server configuration

The RADIUS software used in this example is Juniper Networks Steel-Belted RADIUS Enterprise Edition Version 6.1.2 running on a Windows 2003 server.

When you first log in to Juniper Networks Steel-Belted RADIUS main page, the configuration components are on the left panel as shown in Figure 12-8.

Figure 12-8 Juniper Networks Steel-Belted RADIUS main page

Adding a RADIUS client

IBM Ethernet Switch J48E, as a Network Access Server (NAS), must be added in the RADIUS client entry. Click **RADIUS client** to add IBM Ethernet Switch J48E. Figure 12-9 shows the Edit RADIUS Client window to define the RADIUS Client. You specify the switch name, description, IP address, shared secret key, and model.

Figure 12-9 Adding a RADIUS client

Creating a user for a trusted host

Add a user called `user01` in the existing database for a trusted host. In the Edit Native User window (Figure 12-10), you specify a name, description, and password.

Figure 12-10 Adding user information for user01

To enable dynamic VLAN assignment to the user when authenticated successfully, you must configure the following necessary attributes as illustrated in Figure 12-11:

- Select 802 in Tunnel-Medium-Type.
- Assign 100 in Tunnel-Private-Group-ID (100 is the VLAN ID, which is employee VLAN).
- Select VLAN in Tunnel-Type.

Figure 12-11 Configuring the dynamic VLAN assignment for user01

Creating a user for printer (MAC-based RADIUS authentication)

Add a user using the MAC address of `printer` in the existing database as shown in Figure 12-12. The MAC address of `printer` is the same as the password. However, ensure that you add the password in lowercase, which is mandatory for the authentication to work.

Figure 12-12 Adding user information for printer

To enable dynamic VLAN assignment to `printer` when authenticated successfully, you configure the following required attributes as shown in Figure 12-13:

- Select 802 in Tunnel-Medium-Type.
- Assign 200 in Tunnel-Private-Group-ID (200 is the VLAN ID, which is guest-access VLAN).
- Select VLAN in Tunnel-Type.

Figure 12-13 Configuring the dynamic VLAN assignment for printer

Setting the authentication profiles: Order of Methods

As shown in Figure 12-14, configure user authentication to use EAP-MD5 under Order of Methods in the Authentication Profiles section (shown in Figure 12-8 on page 394).

Figure 12-14 Setting MD5-Challenge

Configuring a trusted host as an 802.1X supplicant

The trusted host is a Windows XP Professional mobile computer. It is configured as an 802.1X supplicant in this example. To configure the trusted host, follow these steps:

1. On the host, open **Network Connections**.
2. Right-click the connection for which you want to enable 802.1x authentication, and click **Properties**.
3. In the Local Area Connection Properties window (Figure 12-15), on the **General** tab, select the **Show icon in notification area when connected** check box.

Figure 12-15 Local Area Connection Properties window

The “balloon” notifications feature is now enabled on Windows XP Professional. This feature provides balloon messages to request a user action for entering the user name and password. It also shows error information for an authentication failure.

4. On the **Authentication** tab, to provide authenticated network access for the Ethernet adapter, select **Enable IEEE 802.1x authentication** as shown in Figure 12-16.

Figure 12-16 Selecting 802.1X authentication

When the Windows XP host is connected to the switch that has port authentication enabled, the devices initiate the port authentication process. After a short time, a balloon message on the task bar prompts you for an authentication action (Figure 12-17).

5. Click the balloon message.

Figure 12-17 Balloon message on the task bar

An Enter Credentials dialog box opens as shown in Figure 12-18.

Figure 12-18 Login window

Configuring IBM Ethernet Switch J48E

Begin with the configuration of the RADIUS server and local NAS attributes that facilitate communication between the authenticator and authentication server:

1. Go to the [edit access] hierarchy to define the RADIUS server parameters as shown in Example 12-35.

Example 12-35 Configuring the RADIUS server parameters

```
{master:0}[edit]
ibm@J48E-VC# edit access <-- Move in the Junos configuration hierarchy

{master:0}[edit access]
ibm@J48E-VC#
set radius-server 10.1.1.8 secret "$9$wT2oGk.569pDi9pOBSys24aDiqmfzn/"
set radius-server 10.1.1.8 retry 5
set radius-server 10.1.1.8 source-address 10.1.1.11

{master:0}[edit access]
ibm@J48E-VC# show
radius-server {
    10.1.1.8 {
        secret "$9$wT2oGk.569pDi9pOBSys24aDiqmfzn/"; ## SECRET-DATA
        retry 5;
        source-address 10.1.1.11;
    }
}
```

2. Specify the IP address of the RADIUS server along with a shared secret key used between the local NAS and RADIUS servers. Set the retry setting to 5 times before the RADIUS server is considered unresponsive and ensure that all RADIUS messages originate from 10.1.1.11.
3. Configure an authentication profile named auth, which is used by authenticator as shown in Example 12-36.

Example 12-36 Configuring the authentication profile

```
{master:0}[edit]
ibm@J48E-VC# edit access <-- Move in the Junos configuration hierarchy
{master:0}[edit access]
ibm@J48E-VC#
set profile auth authentication-order radius
set profile auth radius authentication-server 10.1.1.8

{master:0}[edit access]
ibm@J48E-VC# show
profile auth {
    authentication-order radius;
    radius {
        authentication-server 10.1.1.8;
    }
}
```

4. With the authentication settings in place, configure 802.1X. The 802.1X authenticator properties are configured at the `[edit protocols dot1x]` hierarchy as shown in Example 12-37.

Example 12-37 Configuring 802.1X

```
{master:0}[edit]
ibm@J48E-VC# edit protocols dot1x <-- move in Junos configuration hierarchy

{master:0}[edit protocols dot1x]
ibm@J48E-VC#
set authenticator authentication-profile-name auth
set authenticator interface ge-0/0/1.0 supplicant single-secure
set authenticator interface ge-0/0/1.0 reauthentication 180

set authenticator interface ge-0/0/2.0 supplicant single-secure
set authenticator interface ge-0/0/2.0 mac-radius restrict
set authenticator interface ge-0/0/2.0 reauthentication 180

set authenticator interface ge-0/0/3.0 supplicant single-secure
set authenticator interface ge-0/0/3.0 guest-vlan guest-access
set authenticator interface ge-0/0/3.0 reauthentication 180

{master:0}[edit protocols dot1x]
ibm@J48E-VC# show
authenticator {
    authentication-profile-name auth;
    interface {
        ge-0/0/1.0 {
            supplicant single-secure;
            reauthentication 180;
            guest-vlan guest-access;
        }
        ge-0/0/2.0 {
            supplicant single-secure;
            mac-radius {
                restrict;
            }
            reauthentication 180;
        }
        ge-0/0/3.0 {
            supplicant single-secure;
            reauthentication 180;
            guest-vlan guest-access;
        }
    }
}
```

In this example, you link the authenticator to the already defined `auth` profile. 802.1X is enabled on the ge-0/0/1 interface with single-secure mode. This mode keeps only one active client at any time and forces the client to re-authenticate every 180 seconds.

The ge-0/0/2 interface is set to support MAC-based RADIUS and is restricted to only MAC-based RADIUS by using the restrict keyword. The ge-0/0/3 interface is set to provide limited access in guest-access VLAN for supplicants that fail 802.1X authentication.

Verifying 802.1X authentication

With the RADIUS server, Windows XP 802.1X supplicant, and switch configuration in place, you test the access of the trusted host on the ge-0/0/0 interface.

Testing and verifying the trusted host

To test and verify the trusted host, follow these steps:

1. Connect the Windows XP 8021.X supplicant to the ge-0/0/0 interface. Notice that a balloon message is displayed on the task bar.
2. Click the message.
3. In the login window that opens, enter a valid user name that has been created in the RADIUS server. In this case, you can enter `user01` and a password to authenticate with the RADIUS server.

During communication with the RADIUS server, notice the status of the 802.1X authentication as shown in Example 12-38.

Example 12-38 Viewing the status of 802.1X

```
{master:0}[edit]
ibm@J48E-VC# run show dot1x interface
802.1X Information:
Interface     Role           State           MAC address          User
ge-0/0/1.0    Authenticator  Connecting      00:1C:25:A0:33:22    user01
```

With a valid user name and password, you are granted access. The 802.1X port indicates an *Authenticated* state as shown in Example 12-39.

Example 12-39 802.1X port in authenticated state

```
{master:0}[edit]
ibm@J48E-VC# run show dot1x interface
802.1X Information:
Interface     Role           State           MAC address          User
ge-0/0/1.0    Authenticator  Authenticated   00:1C:25:A0:33:22    user01
```

In addition, you are dynamically assigned a VLAN ID 100 (employee VLAN) as shown in Example 12-40.

Example 12-40 Dynamic VLAN assignment

```
{master:0}[edit]
ibm@J48E-VC# run show dot1x interface detail
ge-0/0/1.0
  Role: Authenticator
  Administrative state: Auto
  Supplicant mode: Single-Secure
  Number of retries: 3
  Quiet period: 60 seconds
  Transmit period: 30 seconds
  Mac Radius: Disabled
  Mac Radius Restrict: Disabled
  Reauthentication: Enabled
  Configured Reauthentication interval: 180 seconds
  Supplicant timeout: 30 seconds
  Server timeout: 30 seconds
```

```
  Maximum EAPOL requests: 2
  Guest VLAN member: <not configured>
  Number of connected supplicants: 1
    Supplicant: user01, 00:1C:25:A0:33:22 <-------------user01 is authenticated
      Operational state: Authenticated
      Authentcation method: Radius <------------- RADIUS authentication is used
      Authenticated VLAN: employee <----- employee VLAN is dynamically assigned
      Session Reauth interval: 180 seconds
      Reauthentication due in 164 seconds
```

Testing and verifying printer

Connect the printer to the ge-0/0/2 interface. Because this authentication is MAC-based RADIUS authentication, you only verify the 802.1X port status as shown in Example 12-41.

Example 12-41 Viewing the MAC-based RADIUS authentication

```
{master:0}[edit protocols dot1x]
ibm@J48E-VC# run show dot1x interface
802.1X Information:
Interface     Role           State          MAC address          User
ge-0/0/2.0    Authenticator  Authenticated  00:1C:25:A0:20:42    001c25a02042
```

You can see that the port is authenticated and the user is the MAC address of the printer. The ge-0/0/2 interface is also configured to dynamically assign VLAN to the printer. To check the assigned VLAN, you can use the **show dot1x interface detail** command as shown in Example 12-42.

Example 12-42 Viewing the details of the MAC-based authentication

```
ibm@J48E-VC# run show dot1x interface ge-0/0/2 detail
ge-0/0/2.0
  Role: Authenticator
  Administrative state: Auto
  Supplicant mode: Single-Secure
  Number of retries: 3
  Quiet period: 60 seconds
  Transmit period: 30 seconds
  Mac Radius: Enabled
  Mac Radius Restrict: Enabled
  Reauthentication: Enabled
  Configured Reauthentication interval: 3600 seconds
  Supplicant timeout: 30 seconds
  Server timeout: 30 seconds
  Maximum EAPOL requests: 2
  Guest VLAN member: <not configured>
  Number of connected supplicants: 1
    Supplicant: 001c25a02042, 00:1C:25:A0:20:42 <---The printer's MAC is
authenticated
      Operational state: Authenticated
      Authentcation method: Mac Radius <-------- MAC RADIUS authentication is used
      Authenticated VLAN: printer <---------- Printer VLAN is dynamically assigned
      Session Reauth interval: 3600 seconds
      Reauthentication due in 3595 seconds
```

The output shows that Restrict MAC-based authentication is used and the `printer` VLAN is assigned.

Testing and verifying the guest

In this example, the guest host does not have a valid user name and password. Therefore, when the guest host is connected to the ge-0/0/3 interface, you do not need to enter a user name. The balloon message box is displayed and prompts you to provide additional information. You can ignore this message. While waiting for the connection, you can see that the state of this 802.1X port has no user to connect as shown in Example 12-43.

Example 12-43 No user to connect

```
{master:0}[edit protocols dot1x]
ibm@J48E-VC# run show dot1x interface
802.1X Information:
Interface     Role           State          MAC address          User
ge-0/0/3.0    Authenticator  Connecting     00:1C:25:A0:20:42    No User
```

With no response from the guest host on the credential, after a while, the switch puts the 802.1X port into the guest-access VLAN as shown in Example 12-44.

Example 12-44 Viewing the guest host details

```
{master:0}[edit protocols dot1x]
ibm@J48E-VC# run show dot1x interface detail
ge-0/0/3.0
  Role: Authenticator
  Administrative state: Auto
  Supplicant mode: Single-Secure
  Number of retries: 3
  Quiet period: 60 seconds
  Transmit period: 30 seconds
  Mac Radius: Disabled
  Mac Radius Restrict: Disabled
  Reauthentication: Enabled
  Configured Reauthentication interval: 180 seconds
  Supplicant timeout: 30 seconds
  Server timeout: 30 seconds
  Maximum EAPOL requests: 2
  Guest VLAN member: guest-access <--------------- Guest-access VLAN is configured
  Number of connected supplicants: 1
    Supplicant: No User, 00:1C:25:A0:20:42 <------------ No user is authenticated
      Operational state: Authenticated
      Authentcation method: GuestVlan <--------- GuestVlan authentication is used
      Authenticated VLAN: guest-access <--------- Guest-access VLAN is dynamically
assigned
      Session Reauth interval: 180 seconds
      Reauthentication due in 153 seconds
```

The output shows that the `GuestVlan` authentication method is used and the `guest-access` VLAN is assigned.

Troubleshooting 802.1X

Junos software provides a useful tool to isolate a problem during 802.1X troubleshooting session. You can enable the 802.1X **traceoption** feature by selecting a few flag options and saving the trace in a file, such as the `dotx-log` file, as shown in Example 12-45.

Example 12-45 Configuring the 802.1X traceoption feature

```
{master:0}[edit]
ibm@J48E-VC# edit protocols dot1x <-- Move in the Junos configuration hierarchy

{master:0}[edit protocols dot1x]
set traceoptions file dot1x-log
set traceoptions flag eapol
set traceoptions flag state
set traceoptions flag dot1x-debug

{master:0}[edit protocols dot1x]
ibm@J48E-VC# show
traceoptions {
    file dot1x-log;
    flag eapol;
    flag state;
    flag dot1x-debug;
}
```

To view the traces in the `dot1x-log` file, you issue the command shown in Example 12-46.

Example 12-46 802.1X log file

```
root@J48E-VC> show log dot1x-log
Feb 22 15:59:52.216626 snmp_epi_register: called
Feb 22 16:00:53.818275  Enabling PNAC on interface 66...

Feb 22 16:00:53.818379  CD Machine called for Port: 66 with Event: PORTENABLED,
State: FORCEBOTH

Feb 22 16:00:53.818406  Successfully exiting CD Machine...

Feb 22 16:00:53.818436  Enabled PNAC on interface 66...

Feb 22 16:00:58.566072  Trying to create a session for MAC: -1c25a0-2042- thro'
Port: 66 ...

Feb 22 16:00:58.566165  Allocated Session Node : 1d7faa8

Feb 22 16:00:58.566224 Generated Unique SessId :9320481655439795134
Feb 22 16:00:58.566252 Acct SessId: 802.1x815900b0 strlen:14
Feb 22 16:00:58.566277  BSM Called with Event: INITIALIZE, and State: Initialise

Feb 22 16:00:58.566306  for Port: 66, MAC: 1c25a0-2042

Feb 22 16:00:58.566332  Id: 0, SessionNode: 1d7faa8

Feb 22 16:00:58.566357  BSM moved to state: INITIALIZE !!

Feb 22 16:00:58.566383  BSM moved to state: IDLE !!
```

```
Feb 22 16:00:58.566414  ASM Called with Event: INITIALIZE, and State: Initialize

Feb 22 16:00:58.566441  for Port: 66, MAC: 1c25a0 - 2042

Feb 22 16:00:58.566467  Id: 0, SessionNode: 1d7faa8
```

If you suspect a malfunction and want to start over, you can issue a **restart dot1x-protocol** operational mode command to restart the 802.1X daemon. Use care with this command because you can globally reset all sessions as indicated by the options shown in Example 12-47.

Example 12-47 Restarting the 802.1X daemon

```
ibm@J48E-VC> restart dot1x-protocol ?
Possible completions:
  <[Enter]>            Execute this command
  gracefully           Gracefully restart the process
  immediately          Immediately restart (SIGKILL) the process
  soft                 Soft reset (SIGHUP) the process
  |                    Pipe through a command
```

Alternatively, you can use the **clear dotx1 interface** command to clear the targeted session. This command resets the hold timers. It is useful for expediting re-authentication attempts after a change is made when you want to quickly see whether it worked.

12.5 More information

For more information about security features, see the following reference materials:

- *System Basics Configuration Guide* for user access and system authentication

  http://www.juniper.net/techpubs/en_US/junos10.1/information-products/topic-collections/config-guide-system-basics/frameset.html

- *Junos OS Policy Framework Configuration Guide* for firewall filters

  http://www-01.ibm.com/support/docview.wss?rs=1337&context=SGPMMT&dc=DA400&uid=isg3T7000157&loc=en_US&cs=utf-8&lang=en

- Port Security on EX Series Switches

  http://www.juniper.net/techpubs/en_US/junos10.1/information-products/pathway-pages/ex-series/port-security.html

- Access Control on EX Series Switches and 802.1X authentication

  http://www.juniper.net/techpubs/en_US/junos10.1/information-products/pathway-pages/ex-series/access-control.html

13

System management

This chapter provides an overview of the network management features of Junos software and explains how to manage networks with Junos software. It also provides information about the basic Simple Network Management Protocol (SNMP) setup. For a more detailed explanation of SNMP, see *JUNOS Software Network Management Configuration Guide*, GA32-0698, at the following address:

http://www-947.ibm.com/systems/support/supportsite.wss/brandmain?brandind=5375876

This chapter includes the following topics:

- Overview of network management
- Overview of the SNMP
- Configuring SNMP
- Overview and configuration of SNMPv3
- MIB support in Junos software
- Using sFlow to monitor on a j-type e-series Ethernet switch
- Configuring system log messages

© Copyright IBM Corp. 2011. All rights reserved.

13.1 Overview of network management

This section provides a basic understanding of the device management functions in Junos software.

13.1.1 Understanding device management functions in Junos software

After you install the device in your network, you must manage the device within the network. Device management can be divided into the following tasks:

- Fault management. Monitor the device; detect and fix faults.
- Configuration management. Configure device attributes.
- Accounting management. Collect statistics for accounting purposes.
- Performance management. Monitor and adjust device performance.
- Security management. Control device access and authenticate users.

The Junos software network management features work with an operations support system to manage the devices within the network. Junos software can assist you in performing these management tasks, as described in Table 13-1.

Table 13-1 Junos software management features

Task	Junos software feature
Fault management	Monitor and see faults using the following features: ► Operational mode commands. For more information about operational mode commands, see the following publications: – *JUNOS System Basics and Services Command Reference,* GA32-0671 – *JUNOS Network Interfaces Command Reference,* GA32-0706 – *JUNOS Routing Protocols and Policies Command Reference,* GA32-0673 ► SNMP Management Information Base (MIB). For more information about the SNMP MIB, see 13.5, “MIB support in Junos software” on page 451. ► Standard SNMP traps. For more information about standard SNMP traps, see *JUNOS Software Network Management Configuration Guide,* GA32-0698. ► Enterprise-specific SNMP traps. For more information about enterprise-specific traps, see *JUNOS Software Network Management Configuration Guide,* GA32-0698. ► System log messages. For more information about configuring system log messages, see the *JUNOS System Basics Configuration Guide,* GA32-0671. For more information about viewing system log messages, see the *JUNOS System Log Messages Reference,* GA32-0675.
Configuration management	► Configure router attributes using the command-line interface (CLI), the JUNOScript API, and the NETCONF API. To configure the router using the CLI, see the *JUNOS System Basics Configuration Guide,* GA32-0671. For more information about configuring the router using the APIs, see the *JUNOScript API Guide,* GA32-0674, and *NETCONF API Guide,* GA32-0678. ► Management Information Base (MIB) for Configuration Management. For more information about the Management Information Base (MIB) for Configuration Management, see *JUNOS Software Network Management Configuration Guide,* GA32-0698.

Task	Junos software feature
Accounting management	Perform the following accounting-related tasks: ► Collect statistics for interfaces, firewall filters, destination classes, source classes, and the Routing Engine. For more information about collecting statistics, see *JUNOS Software Network Management Configuration Guide,* GA32-0698. ► Use interface-specific traffic statistics and other counters, which are available in the Standard Interfaces MIB, Juniper Networks enterprise-specific extensions to the Interfaces MIB, and media-specific MIBs, such as the enterprise-specific ATM MIB. ► Use per-ATM virtual circuit counters, which are available in the enterprise-specific ATM MIB. ► Group source and destination prefixes into source classes and destination classes and count packets for those classes. Collect destination class and source class usage statistics. For more information about classes, see 13.5, “MIB support in Junos software” on page 451. ► Configuring class usage profiles. See the *JUNOS Network Interfaces Configuration Guide,* GA32-0706, and the *JUNOS Policy Framework Configuration Guide,* GA32-0704. ► Count packets as part of a firewall filter. For more information about firewall filter policies, see 13.5, “MIB support in Junos software” on page 451, and the *JUNOS Policy Framework Configuration Guide,* GA32-0704. ► Sample traffic, collect the samples, and send the collection to a host running the CAIDA **`cflowd`** utility. For more information about CAIDA and **`cflowd`**, see the *JUNOS Policy Framework Configuration Guide,* GA32-0704.
Performance management	Monitor performance in the following ways: ► Use operational mode commands. For more information about monitoring performance using operational mode commands, see the *JUNOS System Basics and Services Command Reference,* GA32-0671. ► Use a firewall filter. For more information about performance monitoring using firewall filters, see the *JUNOS Policy Framework Configuration Guide,* GA32-0704. ► Sample traffic, collect the samples, and send the samples to a host running the CAIDA **`cflowd`** utility. For more information about CAIDA and **`cflowd`**, see the *JUNOS Policy Framework Configuration Guide,* GA32-0704. ► Use the enterprise-specific class-of-service MIB. For more information about this MIB, see 13.5, “MIB support in Junos software” on page 451.
Security management	Assure security in your network in the following ways: ► Control access to the router and authenticate users. For more information about access control and user authentication, see the *JUNOS System Basics Configuration Guide,* GA32-0671. ► Control access to the router using SNMPv3 and SNMP over IPv6. For more information, see 13.4.3, “Configuring the local engine ID” on page 428, and 13.3.12, “Tracing SNMP activity on a device running Junos software” on page 421.

13.2 Overview of the SNMP

The SNMP enables the monitoring of network devices from a central location. This topic provides an overview of SNMP and describes how SNMP is implemented in Junos software.

This topic includes the following sections:

- Architecture of the SNMP
- Features of the Junos SNMP agent
- Management Information Base
- SNMP traps and informs

13.2.1 Architecture of the SNMP

The SNMP agent exchanges network management information with the SNMP manager software running on a network management station (NMS) or host. The agent responds to requests for information and actions from the manager. The agent also controls access to the MIB of the agent, which is the collection of objects that can be viewed or changed by the SNMP manager.

The SNMP manager collects information about network connectivity, activity, and events by polling managed devices.

Communication between the agent and the manager occurs in one of the following forms:

- **`Get`**, **`GetBulk`**, and **`GetNext`** requests

  The manager requests information from the agent. The agent returns the information in a Get response message.

- **`Set`** requests

  The manager changes the value of an MIB object controlled by the agent. The agent indicates the status in a Set response message.

- **`Traps`** notification

  The agent sends traps to notify the manager of significant events that occur on the network device.

13.2.2 Features of the Junos SNMP agent

The Junos SNMP agent software consists of an SNMP master agent that delegates all SNMP requests to subagents. Each subagent is responsible for the support of a specific set of MIBs.

Junos software supports the following versions of the SNMP:

SNMPv1 The initial implementation of SNMP that defines the architecture and framework for SNMP.

SNMPv2c The revised protocol, with improvements to performance and manager-to-manager communications. Specifically, SNMPv2c implements community strings, which act as passwords when determining who, what, and how the SNMP clients can access the data in the SNMP agent. The community string is contained in SNMP **`get`**, **`getBulk`**, **`getNext`**, and **`Set`** requests. The agent might require a different community string for **`get`**, **`getBulk`**, and **`getNext`** requests (read-only access) than it does for Set requests (read-write access).

SNMPv3 The most up-to-date protocol with a focus on security. SNMPv3 defines a security model, user-based security model (USM), and a view-based access control model (VACM). SNMPv3 USM provides data integrity, data origin authentication, message replay protection, and protection against disclosure of the message payload. SNMPv3 VACM provides access control to determine whether a specific type of access (read or write) to the management information is allowed.

In addition, the Junos SNMP agent software accepts IPv4 and IPv6 addresses for transport over IPv4 and IPv6. For IPv6, Junos software supports the following IPv6 over SNMP:

- SNMP data over IPv6 networks
- IPv6-specific MIB data
- SNMP agents for IPv6

13.2.3 Management Information Base

An MIB is a hierarchy of information used to define managed objects in a network device. The MIB structure is based on a tree structure, which defines a grouping of objects into related sets. Each object in the MIB is associated with an object identifier (OID), which names the object. The leaf in the tree structure is the managed object instance, which represents a resource, event, or activity that occurs in the network device.

MIBs are either standard or enterprise-specific. Standard MIBs are created by the Internet Engineering Task Force (IETF) and documented in various RFCs. Depending on the vendor, many standard MIBs are delivered with the NMS software. You can also download the standard MIBs from the following IETF website, and compile them into your NMS, if necessary:

http://www.ietf.org

Enterprise-specific MIBs are developed and supported by a specific equipment manufacturer. If your network contains devices that have enterprise-specific MIBs, you must obtain them from the manufacturer and compile them into your network management software.

For a list of standard supported MIBs and Juniper Networks enterprise-specific supported MIBs, see 13.5, “MIB support in Junos software” on page 451.

13.2.4 SNMP traps and informs

Routers can send notifications to SNMP managers when significant events occur on a network device, which most often are errors or failures. SNMP notifications can be sent as traps or inform requests. SNMP traps are unconfirmed notifications, and SNMP informs are confirmed notifications.

SNMP traps are defined in either standard or enterprise-specific MIBs. Standard traps are created by the IETF and documented in various RFCs. The standard traps are compiled into the network management software. You can also download the standard traps from the IETF website at the following address:

http://www.ietf.org

For more information about standard traps supported by Junos software, see *JUNOS Software Network Management Configuration Guide,* GA32-0698.

Enterprise-specific traps are developed and supported by a specific equipment manufacturer. If your network contains devices that have enterprise-specific traps, you must obtain them from the manufacturer and compile them into your network management software.

For more information about enterprise-specific traps supported by Junos software, see *JUNOS Software Network Management Configuration Guide,* GA32-0698. For information about system logging severity levels for SNMP traps, see “System logging severity levels for SNMP traps” on page 414.

With traps, the receiver does not send any acknowledgment when it receives a trap, and the sender cannot determine if the trap was received. To increase reliability, SNMP informs are supported in SNMPv3. An SNMP manager that receives an inform acknowledges the message with a response. For information about SNMP informs, see 13.4.18, “Configuring SNMP informs” on page 447.

SNMP trap queuing

Junos software supports trap queuing to ensure that traps are not lost because of temporary unavailability of routes. Two types of queues, destination queues and a throttle queue, are formed to ensure delivery of traps and control the trap traffic.

Junos software forms a destination queue when a trap to a particular destination is returned because the host is not reachable. It adds the subsequent traps to the same destination to the queue. Junos software checks for availability of routes every 30 seconds and sends the traps from the destination queue in a round-robin fashion.

If the trap delivery fails, the trap is added back to the queue. Then the delivery attempt counter and the next delivery attempt timer for the queue are reset. Subsequent attempts occur at progressive intervals of 1 minute, 2 minutes, 4 minutes, and 8 minutes. The maximum delay between the attempts is 8 minutes, and the maximum number of attempts is 10. After 10 unsuccessful attempts, the destination queue and all the traps in the queue are deleted.

Junos software also has a throttle mechanism. This mechanism controls the number of traps (**`throttle threshold`**; default value of 500 traps) that are sent during a particular time period (**`throttle interval`**; default of 5 seconds). It also ensures consistency in trap traffic, especially when large number of traps are generated because of interface status changes. The throttle interval period begins when the first trap arrives at the throttle. All traps within the trap threshold are processed, and the traps beyond the threshold limit are queued. The maximum size of the throttle queue is 50k. When a trap is added to the throttle queue, or if the throttle queue has exceeded the maximum size, the trap is added back on top of the destination queue. Then all subsequent attempts from the destination queue are stopped for a 30 second period, after which the destination queue restarts sending the traps.

Trap queuing in Junos software: You cannot configure Junos software for trap queuing. Nor can you view any information about trap queues except what is available in the syslog.

System logging severity levels for SNMP traps

For some traps, when a trap condition occurs, the trap is logged if the system logging is configured to log an event with that system logging severity level. This logging happens regardless of whether the SNMP agent sends a trap to an NMS. For more information about system logging severity levels, see the *JUNOS Software System Basics Configuration Guide,* GA32-0671. For more information about system logging severity levels for standard traps, see *JUNOS Software Network Management Configuration Guide,* GA32-0698. For more information about system logging severity levels for enterprise-specific traps, see *JUNOS Software Network Management Configuration Guide,* GA32-0698.

13.3 Configuring SNMP

This chapter contains the following topics:

- Configuring SNMP on a device running Junos software
- Configuring the system contact on a device running Junos software
- Configuring the system location for a device running Junos software
- Configuring the description on a device running Junos software
- Filtering duplicate SNMP requests
- Configuring the commit delay timer
- Configuring the system name
- Configuring the SNMP community string and options

- Configuring the SNMP trap options
- Configuring SNMP trap groups
- Configuring MIB views
- Tracing SNMP activity on a device running Junos software

13.3.1 Configuring SNMP on a device running Junos software

By default, SNMP is disabled on devices running Junos software. To enable SNMP on a router or switch, you must include the SNMP configuration statements at the `[edit snmp]` hierarchy level.

13.3.2 Configuring the system contact on a device running Junos software

You can specify an administrative contact for each system that is managed by the SNMP. This name is placed into the MIB II `sysContact` object. To configure a contact name, include the **`contact`** statement at the `[edit snmp]` hierarchy level, enter the following command:

```
[edit snmp]
root# set contact contact name
```

If the name contains spaces, enclose it in quotation marks (" ").

To define a system contact name that contains spaces, enter the following command:

```
[edit snmp]
root# set contact "Juniper Berry, (650) 555-1234"
```

13.3.3 Configuring the system location for a device running Junos software

You can specify the location of each system that is managed by the SNMP. This string is placed into the MIB II `sysLocation` object. To configure a system location, include the **`location`** statement at the `[edit snmp]` hierarchy level:

```
[edit snmp]
set location location
```

If the location contains spaces, enclose it in quotation marks (" ").

To specify the system location, enter the following command:

```
[edit snmp]
root# set location "Row 11, Rack C"
```

13.3.4 Configuring the description on a device running Junos software

You can specify a description for each system that is managed by the SNMP. This string is placed into the MIB II `sysDescription` object. To configure a description, include the **`description`** statement at the `[edit snmp]` hierarchy level:

```
[edit snmp]
root# set description description
```

If the description contains spaces, enclose it in quotation marks (" ").

To specify the system description, enter the following command:

```
[edit snmp]
root# set description "M40 router with 8 FPCs"
```

13.3.5 Filtering duplicate SNMP requests

By default, filtering duplicate **get**, **getNext**, and **getBulk** SNMP requests is disabled on devices that are running Junos software. If an NMS retransmits a **get**, **getNext**, or **getBulk** SNMP request too frequently to the router, the request might interfere with the processing of previous requests. It might also slow down the response time of the agent. Filtering these duplicate requests improves the response time of the SNMP agent. Junos software uses the following information to determine if an SNMP request is a duplicate:

- Source IP address of the SNMP request
- Source User Datagram Protocol (UDP) port of the SNMP request
- Request ID of the SNMP request

To filter duplicate SNMP requests, include the **filter-duplicates** statement at the [edit snmp] hierarchy level:

```
[edit snmp]
root# set filter-duplicates
```

13.3.6 Configuring the commit delay timer

When a router or switch first receives an SNMP nonvolatile **Set** request, a JUNOScript session opens and prevents other users or applications from changing the candidate configuration (equivalent to the CLI **configure exclusive** command). If the router does not receive new SNMP **Set** requests within 5 seconds (the default value), the candidate configuration is committed and the JUNOScript session closes. (The configuration lock is released.) If the router receives new SNMP **Set** requests while the candidate configuration is being committed, the SNMP **Set** request is rejected and an error is generated. If the router receives new SNMP Set requests before 5 seconds have lapsed, the commit-delay timer resets to 5 seconds. (The commit-delay timer defines the length of time between when the last SNMP request is received and the commit is requested.) By default, the commit-delay timer is set to 5 seconds.

To configure the commit-delay timer for the SNMP **Set** reply and start of the commit, include the **commit-delay** statement at the [edit snmp nonvolatile] hierarchy level:

```
[edit snmp nonvolatile]
root# set commit-delay seconds
```

where *seconds* is the length of the time between when the SNMP request is received and the commit is requested for the candidate configuration.

For more information about the **configure exclusive** command and locking the configuration, see the *JUNOS Software CLI User Guide,* GA32-0697.

13.3.7 Configuring the system name

With the Junos software, you can override the system name by including the **name** statement at the [edit snmp] hierarchy level:

```
[edit snmp]
root# set name name
```

If the name contains spaces, enclose it in quotation marks (" ").

13.3.8 Configuring the SNMP community string and options

The SNMP community string defines the relationship between an SNMP server system and the client systems. This string acts similar to a password to control the access of clients to the server. To configure a community string in a Junos software configuration, include the **community** statement at the [edit snmp] hierarchy level:

```
[edit snmp]
root# set community name
```

If the community name contains spaces, enclose it in quotation marks (" ").

The default authorization level for a community is read-only. To allow Set requests within a community, you must define that community as authorization read-write. For Set requests, you must also include the specific MIB objects that are accessible with read-write privileges using the view statement. To configure the authorization level in a Junos software configuration, include the **authorization** statement at the [edit snmp community name] hierarchy level:

```
[edit snmp community name]
root# set authorization [read only | read write]
```

The **clients** statement lists the IP addresses of the clients (community members) that are allowed to use this community. If no clients statement is present, all clients are allowed. For address, you must specify an IPv4 or IPv6 address, not a host name:

```
[edit snmp community name]
root# set clients address
```

Community names: Community names must be unique. You cannot configure the same community name at the [edit snmp community] and [edit snmp v3 snmp-community community-index] hierarchy levels.

13.3.9 Configuring the SNMP trap options

By using the SNMP trap options, you can set the source address of every SNMP trap packet that is sent by the router to a single address regardless of the outgoing interface. In addition, you can set the agent address of the SNMPv1 traps.

SNMP and routing instances: SNMP cannot be associated with any routing instances other than the master routing instance.

To configure SNMP trap options, include the **trap-options** statement at the [edit snmp] hierarchy level:

```
[edit snmp]
root# edit trap-options
[edit snmp trap options]
root# set agent-address outgoing-interface
root# set source-address address
```

You must also configure a trap group for the trap options to take effect. For information about trap groups, see 13.3.10, “Configuring SNMP trap groups” on page 418.

Configuring the source address for SNMP traps

You can configure the source address of trap packets in two ways: lo0 or a valid IPv4 address configured on one of the router interfaces. The lo0 value indicates that the source address of the SNMP trap packets is set to the lowest loopback address configured on the lo0 interface.

To specify a valid interface address as the source address for SNMP traps on one of the router interfaces, include the **source-address** statement at the [edit snmp trap-options] hierarchy level:

```
[edit snmp trap-options]
root# set source-address address
```

where *address* is a valid IPv4 address configured on one of the router interfaces.

To specify the source address of the SNMP traps so that they are sent to the lowest loopback address configured on the lo0 interface, include the **source-address** statement at the [edit snmp trap-options] hierarchy level:

```
[edit snmp trap-options]
root# set source-address lo0;
```

Configuring the agent address for SNMP traps

The agent address is only available in SNMPv1 trap packets. By default, the default local address of the router is used in the agent address field of the SNMPv1 trap. To configure the agent address, include the **agent-address** statement at the [edit snmp trap-options] hierarchy level. Currently, the agent address can only be the address of the outgoing interface:

```
[edit snmp trap-options]
root# set agent-address outgoing-interface
```

13.3.10 Configuring SNMP trap groups

You can create and name a group of one or more types of SNMP traps and then define which systems receive the group of SNMP traps. The trap group must be configured for SNMP traps to be sent. To create an SNMP trap group, include the **trap-group** statement at the [edit snmp] hierarchy level:

```
[edit snmp]
root# set trap-group group-name
[edit snmp trap-group group-name]
root# set categories category
root# set destination-port port-number
root# set routing-instance instance
root# set targets address
root# set version [all | v1 | v2]
```

The trap group name can be any string and is embedded in the community name field of the trap. To configure your own trap group port, include the **`destination-port`** statement. The default destination port is port 162.

For each trap group that you define, you must include the **`target`** statement to define at least one system as the recipient of the SNMP traps in the trap group. Specify the IPv4 or IPv6 address of each recipient, not its host name.

Specify the types of traps that the trap group can receive in the **`categories`** statement. A trap group can receive but is not limited to the following categories:

authentication Authentication failures.

chassis Chassis or environment notifications.

configuration Configuration notifications.

link Link-related notifications (up-down transitions, DS-3 and DS-1 line status change, IPv6 interface state change, and Passive Monitoring PIC overload).

> **Passive Monitoring PIC overload interface traps:** To send Passive Monitoring PIC overload interface traps, select the **link** trap category.

remote-operations Remote operation notifications.

rmon-alarm Alarm for RMON events.

routing Routing protocol notifications.

startup System warm and cold starts.

vrrp-events Virtual Router Redundancy Protocol (VRRP) events, such as new-master or authentication failures.

For a full list of categories, see *JUNOS Software Network Management Configuration Guide,* GA32-0698.

With the **`version`** statement, you can specify the SNMP version of the traps that are sent to targets of the trap group. If you specify `v1` only, SNMPv1 traps are sent. If you specify `v2` only, SNMPv2 traps are sent. If you specify `all`, both an SNMPv1 and an SNMPv2 trap are sent for every trap condition.

Example: Configuring SNMP trap groups

Example 13-1 shows how to set up a trap notification list named *urgent-dispatcher* for link and startup traps. This list is used to identify the network management hosts (1.2.3.4 and fe80::1:2:3:4) to which traps generated by the local router must be sent. The name specified for a trap group is used as the SNMP community string when the agent sends traps to the listed targets.

Example 13-1 Configuring SNMP trap groups

```
[edit]
root# edit snmp

[edit snmp]
root# set trap-group "urgent-dispatcher"

[edit snmp]
```

```
root# edit trap-group "urgent-dispatcher"

[edit snmp trap-group urgent-dispatcher]
root# set version v2

[edit snmp trap-group urgent-dispatcher]
root# set catagories link startup

[edit snmp trap-group urgent-dispatcher]
root# set targets 1.2.3.4

[edit snmp trap-group urgent-dispatcher]
root# set targets fe80::1:2:3:4

[edit snmp trap-group urgent-dispatcher]
root# top

[edit]
root# show snmp
trap-options;
trap-group urgent-dispatcher {
    version v2;
    categories {
        link;
        startup;
    }
    targets {
        1.2.3.4;
        fe80::1:2:3:4;
    }
}

[edit]
root#
```

13.3.11 Configuring MIB views

By default, an SNMP community grants read access and denies write access to all supported MIB objects, even communities configured as authorization read-write. To restrict or grant read or write access to a set of MIB objects, you must configure a MIB view and associate the view with a community. To configure MIB views, include the **view** statement at the [edit snmp] hierarchy level:

```
[edit snmp]
root# set view view-name oid object-identifier [include | exclude]
```

The **view** statement defines an MIB view and identifies a group of MIB objects. Each MIB object of a view has a common OID prefix. Each OID represents a subtree of the MIB object hierarchy. The subtree can be represented by either a sequence of dotted integers (such as 1.3.6.1.2.1.2) or by its subtree name (such as interfaces). A configuration statement uses a view to specify a group of MIB objects on which to define access. You can also use the wildcard character asterisk (*) to include OIDs that match a particular pattern in the SNMP view. To enable a view, you must associate the view with a community.

Removing an OID: To remove an OID completely, use the **delete view all oid oid-number** command, but omit the include parameter.

To associate MIB views with a community, include the **view** statement at the [edit snmp community *community-name*] hierarchy level:

```
[edit snmp community community-name]
root# set view view-name
```

13.3.12 Tracing SNMP activity on a device running Junos software

SNMP tracing operations track activity for SNMP agents and record the information in log files. The logged error descriptions provide detailed information to help you solve problems faster.

By default, Junos software does not trace any SNMP activity. If you include the **traceoptions** statement at the [edit snmp] hierarchy level, the default tracing behavior is that important activities are logged in files in the /var/log directory. Each log is named after the SNMP agent that generates it. Currently, the following log files are created in the /var/log directory when the **traceoptions** statement is used:

- chassisd
- craftd
- ilmid
- mib2d
- rmopd
- serviced
- snmpd

When a trace file named *filename* reaches its maximum size, it is renamed *filename.0*, then *filename.1*, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten. For more information about how log files are created, see the *JUNOS System Log Messages Reference,* GA32-0675.

13.3.13 Configuring the SNMP log file

Log files can be accessed only by the user who configures the tracing operation. You cannot change the /var/log directory in which trace files are located. However, you can customize the other trace file settings by including the following statements at the [edit snmp traceoptions] hierarchy level:

```
[edit snmp traceoptions]
root# set file [files number between 2...1000] [match regular-expression] [size
max size between 10000..1073741824] [world-readable | no-world-readable]
[edit snmp traceoptions]
root# set flag flag
[edit snmp traceoptions]
root# set no-remote-trace
```

These statements are described in the following sections:

- Configuring the number and size of the SNMP log files
- Configuring access to the log file
- Configuring a regular expression for lines to be logged
- Configuring the trace operations

Configuring the number and size of the SNMP log files

By default, when the trace file, `filename`, reaches 128 KB in size, it is renamed `filename.0`, then `filename.1`, and so on, until there are three trace files. Then the oldest trace file, `filename.2`, is overwritten.

You can configure the limits on the number and size of trace files by including the following statements at the `[edit snmp traceoptions]` hierarchy level:

```
[edit snmp traceoptions]
root# set file files number size size
```

For example, set the maximum file size to 2 MB, and the maximum number of files to 20 MB. When the file that receives the output of the tracing operation `filename` reaches 2 MB, the *filename* file is renamed `filename.0`, and a new file called `filename` is created. When the new `filename` file reaches 2 MB, `filename.0` is renamed `filename.1`, `filename` is renamed `filename.0`, and a new file called `filename` is created. This process repeats until there are 20 trace files. Then the oldest file `filename.19` is overwritten by the newest `filename` file.

The number of files can be in the range 2–1000 files. The file size of each file can be 10 KB –1 GB.

Configuring access to the log file

By default, log files can be accessed only by the user who configures the tracing operation. To specify that any user can read all log files, include the **file world-readable** statement at the `[edit snmp traceoptions]` hierarchy level:

```
[edit snmp traceoptions]
root# set file world-readable
```

To explicitly set the default behavior, include the **file no-world-readable** statement at the `[edit snmp traceoptions]` hierarchy level:

```
[edit snmp traceoptions]
root# set file no-world-readable;
```

Configuring a regular expression for lines to be logged

By default, the trace operation output includes all lines that are relevant to the logged activities. You can refine the output by including the **match** statement at the `[edit snmp traceoptions file filename]` hierarchy level and specifying a regular expression `regex` to be matched:

```
[edit snmp traceoptions]
root# set file filename match regular-expression
```

Configuring the trace operations

By default, only important activities are logged. You can specify which trace operations are to be logged by including the following **flag** statement, with one or more tracing flags, at the `[edit snmp traceoptions]` hierarchy level:

```
[edit snmp traceoptions]
root# set flag [all] [configuration] [database] [events] [general]
[interface-stats] [nonvolotile-sets] [pdu] [policy] [protocol-timeouts]
[routing-socket] [server] [subagent] [timer] [varbind-error]
```

Table 13-2 explains the meaning of the SNMP tracing flags.

Table 13-2 SNMP tracing flags

Flag	Description	Default setting
all	Log all operations.	Off
configuration	Log reading of configuration at the `[edit snmp]` hierarchy level.	Off
database	Log events involving storage and retrieval in events database.	Off
events	Log important events.	Off
general	Log general events.	Off
interface-stats	Log physical and logical interface statistics.	Off
nonvolatile-sets	Log nonvolatile SNMP set request handling.	Off
pdu	Log SNMP request and response packets.	Off
policy	Log policy processing.	Off
protocol-timeouts	Log SNMP response timeouts.	Off
routing-socket	Log routing socket calls.	Off
server	Log communication with processes that are generating events.	Off
subagent	Log subagent restarts.	Off
timer	Log internal timer events.	Off
varbind-error	Log variable binding errors.	Off

To view the end of the log for an agent, enter the **`show log`** *`agent`***`d | last`** operational mode command:

```
[edit]
root# run show log agentd | last
```

where *`agent`* is the name of an SNMP agent.

13.4 Overview and configuration of SNMPv3

In contrast to SNMPv1 and SNMPv2, SNMPv3 supports authentication and encryption. SNMPv3 uses the USM for message security and the VACM for access control. The USM specifies authentication and encryption. The VACM specifies access-control rules.

The USM uses the concept of a user for which security parameters, levels of security, authentication, privacy protocols, and keys are configured for both the agent and the manager. Messages that are sent by using the USM are better protected than messages that are sent with community strings, where passwords are sent in the clear. With the USM, messages that are exchanged between the manager and the agent can have data integrity checking and data origin authentication. The USM protects against message delays and message replays by using time indicators and request IDs. Encryption is also available.

To complement the USM, SNMPv3 uses the VACM, which is a highly granular access-control model for SNMPv3 applications. Based on the concept of applying security policies to the name of the groups querying the agent, the agent decides whether the group is allowed to view or change specific MIB objects. The VACM defines collections of data, called *views*, groups of data users, and access statements that define which views a particular group of users can use for reading, writing, or receiving traps.

Trap entries in SNMPv3 are created by configuring the `notify`, `notify filter`, `target address`, and `target` parameters. The **`notify`** statement specifies the type of notification and contains a single tag. The tag defines a set of target addresses to receive a trap. The notify filter defines access to a collection of trap OIDs. The target address defines the address of a management application and other attributes to use in sending notifications. Target parameters define the message processing and security parameters to be used in sending notifications to a particular management target.

This topic includes the following sections:

- SNMPv3 configuration statements
- Minimum SNMPv3 configuration on a device running Junos software
- Configuring the local engine ID
- Creating SNMPv3 users
- Configuring the SNMPv3 authentication type
- Configuring the encryption type
- Example: Defining SNMPv3 users
- Defining access privileges for an SNMP group
- Configuring the access privileges granted to a group
- Example: Defining access privileges
- Assigning security names to groups
- Configuring SNMPv3 traps on a device running Junos software
- Configuring the SNMPv3 trap notification
- Configuring the trap notification filter
- Configuring the trap target address
- Example: Configuring the tag list
- Defining and configuring the trap target parameters
- Configuring SNMP informs
- Configuring the remote engine and remote user
- Configuring the inform notification type and target address
- Configuring the SNMPv3 community

13.4.1 SNMPv3 configuration statements

To configure SNMPv3, include the following statements at the `[edit snmp v3]` and `[edit snmp]` hierarchy levels. Example 13-2 shows the SNMPv3 configuration statements.

Example 13-2 SNMPv3 configuration statements

```
[edit snmp]
root# set engine-id [local engine-id | use-fxp0-mac-address |
use-default-ip-address]

[edit snmp]
root# set view view-name oid object-identifier [include | exclude]

[edit snmp v3]
root# set notify name tag tag-name type [trap | inform])
```

```
[edit snmp v3]
root# set notify-filter profile-name oid object-identifier [include | exclude]

[edit snmp v3]
root# set snmp-community community-index community-name community-name
security-name security-name tag tag-name

[edit snmp v3]
root# target-address target-address-name address address address-mask address-mask
inform-retry-count number inform-timeout seconds port port-number routing-instance
instance tag-list tag-list target-parameters target-parameters-name

[edit snmp v3]
root# set target-parameters target-parameters-name notify-filter profile-name

[edit snmp v3]
root# set target-parameters target-parameters-name parameters
message-processing-model [v1 | v2c | v3] security-model [usm | v1 | v2c]
security-level [authentication | none | privacy] security-name security-name

[edit snmp v3]
root# set usm [local-engine | remote-engine engine-id] user username
authentication-md5 authentication-password authentication-password
or
authentication-none
or
authentication-sha authentication-password authentication-password
or
privacy-3des privacy-password privacy-password
or
privacy-aes128 privacy-password privacy-password
or
privacy-des privacy-password privacy-password
or
privacy-none

[edit snmp v3]
root# set vacm access group group-name default-context-prefix security-model [any
| usm | v1 | v2c] security-level [authentication | none | privacy] notify-view
view-name read-view view-name write-view view-name

[edit snmp v3]
root# security-to-group security-model [usm | v1 | v2c] security-name
security-name group group-name
```

13.4.2 Minimum SNMPv3 configuration on a device running Junos software

Example 13-3 shows the minimum statements that are required to configure SNMP on the IBM j-type e-series Ethernet switches and IBM j-type m-series Ethernet routers. These statements are included at the [edit snmp v3] and [edit snmp] hierarchy levels of the Junos software configuration.

Example 13-3 Minimum SNMPv3 configuration

```
[edit]
root# edit snmp

[edit snmp]
root# set view vi oid interfaces include

[edit snmp]
root# edit v3

[edit snmp v3]
root# set notify n1 tag t1

[edit snmp]
root# set notify-filter prof1 oid interfaces include

[edit snmp v3]
root# set snmp-community com1 security-name sec1

[edit snmp v3]
root# set target-address targ1 address 10.1.1.204 target-parameters tparm1

[edit snmp v3]
root# set target-parameters tparm1 notify-filter prof1

[edit snmp v3]
root# set target-parameters tparm1 parameters message-processing-model v3
security-model usm security-level none security-name sec1

[edit snmp v3]
root# set usm local-engine user ibm privacy-none authentication-none

[edit snmp v3]
root# set vacm access group grp1 default-context-prefix security-model any
security-level none notify-view nv1 write-view wv1 read-view rv1


[edit snmp v3]
root# edit vacm

[edit snmp v3 vacm]
root# ...security-model usm security-name sec1 group grp1

[edit snmp v3 vacm]
root# up

[edit snmp v3]
root# up
```

```
[edit snmp]
root# show
v3 {
   usm {
      local-engine {
         user ibm {
            authentication-none;
            privacy-none;
         }
      }
   }
   vacm {
      security-to-group {
         security-model usm {
            security-name sec1 {
               group grp1;
         }
      }
   }
   access {
      group grp1 {
         default-context-prefix {
            security-model any {
               security-level none {
                  read-view rv1;
                  write-view wv1;
                  notify-view nv1;
               }
            }
         }
      }
   }
   target-address targ1 {
      address 10.1.1.204;
         target-parameters tparm1;
   }
   target-parameters tparm1 {
      parameters {
         message-processing-model v3;
         security-model usm;
         security-level none;
         security-name sec1;
      }
      notify-filter prof1;
   }
   notify n1 {
      tag t1;
   }
   notify-filter prof1 {
      oid interfaces include;
   }
   snmp-community com1 {
      security-name sec1;
```

```
   }
}
view vi {
   oid interfaces include;
}
```

Configuration at the [edit snmpview-name] hierarchy level: You must configure at least one view, notify, read, or write at the `[edit snmpview-name]` hierarchy level.

13.4.3 Configuring the local engine ID

By default, the local engine ID uses the default IP address of the router. The local engine ID is the administratively unique identifier for the SNMPv3 engine. This statement is optional. To configure the local engine ID, include the **`engine-id`** statement at the `[edit snmp]` hierarchy level:

```
[edit snmp]
root# set engine-id [local engine-id-suffix | use-default-ip-address |
use-mac-address]
```

where:

`local` *`engine-id-suffix`*	is the engine ID suffix that is explicitly configured.
`use-default-ip-address`	is the engine ID suffix that is generated from the default IP address.
`use-mac-address`	is the SNMP engine identifier that is generated from the Media Access Control (MAC) address of the management interface on the router.

The local engine ID is defined as the administratively unique identifier of an SNMPv3 engine and is used for identification, not for addressing. An engine ID has two portions: prefix and suffix. The prefix is formatted according to the specifications defined in RFC 3411, An Architecture for Describing SNMP Management Frameworks. RFC details are available at the following website:

http://www.ietf.org

You can only configure the suffix portion here.

SNMPv3 authentication and encryption keys: SNMPv3 authentication and encryption keys are generated based on the associated passwords and the engine ID. If you configure or change the engine ID, you must commit the new engine ID before you configure SNMPv3 users. Otherwise the keys generated from the configured passwords are based on the previous engine ID. For the engine ID, use the IP address of the device. Alternatively, you can use the MAC address of *fxp0* if the device has only one Routing Engine.

13.4.4 Creating SNMPv3 users

For each SNMPv3 user, you can specify the *user name*, *authentication type*, *authentication password*, *privacy type*, and *privacy password*. After the password is entered, a key based on the engine ID and password is generated and written to the configuration file. After key generation, the password is deleted from this file.

Restriction: You can only configure one encryption type for each SNMPv3 user.

To create users, include the user statement at the [`edit snmp v3 usm local-engine`] hierarchy level:

```
[edit snmp v3 usm local-engine]
root# set user username
```

where *`username`* is the name that identifies the SNMPv3 user.

To configure user authentication and encryption, include the following statements at the [`edit snmp v3 usm local-engine user username`] hierarchy level:

```
[edit snmp v3 usm local-engine user username]
root# set authentication-md5 authentication-password authentication-password
```

Alternatively, you can use any of the following commands after `root#`:

```
set authentication-sha authentication-password authentication-password
set authentication-none
set privacy-aes128 privacy-password privacy-password
set privacy-des privacy-password privacy-password
set privacy-3des privacy-password privacy-password
set privacy-none
```

13.4.5 Configuring the SNMPv3 authentication type

By default, in a Junos software configuration, the SNMPv3 authentication type is set to none. This topic includes the following sections:

- Configuring MD5 authentication
- Configuring Secure Hash Algorithm authentication
- Configuring no authentication

Configuring MD5 authentication

To configure the message-digest algorithm 5 (MD5) as the authentication type for a SNMPv3 user, include the **`authentication-md5`** statement at the [`edit snmp v3 usm local-engine user username`] hierarchy level:

```
[edit snmp v3 usm local-engine user username]
root# set authentication-md5 authentication-password authentication-password
```

where *`authentication-password`* is the password used to generate the key used for authentication.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Configuring Secure Hash Algorithm authentication

To configure the Secure Hash Algorithm (SHA) as the authentication type for an SNMPv3 user, include the **`authentication-sha`** statement at the `[edit snmp v3 usm local-engine user username]` hierarchy level:

```
[edit snmp v3 usm local-engine user username]
root# set authentication-sha authentication-password authentication-password}
```

where *`authentication-password`* is the password used to generate the key that is used for authentication.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Configuring no authentication

To configure no authentication for an SNMPv3 user, include the **`authentication-none`** statement at the `[edit snmp v3 usm local-engine user username]` hierarchy level:

```
[edit snmp v3 usm local-engine user username]
root# set authentication-none
```

13.4.6 Configuring the encryption type

By default, encryption is set to none. This topic includes the following sections:

- Configuring the Advanced Encryption Standard algorithm
- Configuring the Data Encryption Standard algorithm
- Configuring the Triple DES algorithm
- Configuring no encryption

> **Prerequisite for encryption:** Before you configure encryption, you must configure MD5 or SHA authentication. Before you configure the **`privacy-3des`** and **`privacy-aes128`** statements, you must install the `jcrypto` package.

Configuring the Advanced Encryption Standard algorithm

To configure the Advanced Encryption Standard (AES) algorithm for an SNMPv3 user, include the **`privacy-aes128`** statement at the `[edit snmp v3 usm local-engine user username]` hierarchy level:

```
[edit snmp v3 usm local-engine user username]
root# set privacy-aes128 privacy-password privacy-password
```

where *`privacy-password`* is the password that is used to generate the key used for encryption.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Configuring the Data Encryption Standard algorithm

To configure the Data Encryption Standard (DES) algorithm for an SNMPv3 user, include the **privacy-des** statement at the [edit snmp v3 usm local-engine user *username*] hierarchy level:

```
[edit snmp v3 usm local-engine user username]
root# set privacy-des privacy-password privacy-password
```

where *privacy-password* is the password that is used to generate the key used for encryption.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Configuring the Triple DES algorithm

To configure the Triple DES algorithm for an SNMPv3 user, include the **privacy-3des** statement at the [edit snmp v3 usm local-engine user *username*] hierarchy level:

```
[snmp v3 usm local-engine user username]
root# set privacy-3des privacy-password privacy-password
```

where *privacy-password* is the password that is used to generate the key used for encryption.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Configuring no encryption

To configure no encryption for an SNMPv3 user, include the **privacy-none** statement at the [edit snmp v3 usm local-engine user *username*] hierarchy level:

```
[edit snmp v3 usm local-engine user username]
root# set privacy-none
```

13.4.7 Example: Defining SNMPv3 users

Example 13-4 shows the configuration steps for defining SNMPv3 users.

Example 13-4 Defining SNMPv3 users

```
[edit]
root# edit snmp v3

[edit snmp v3]
root# set usm local-engine user username1 authentication-md5
authentication-password password1

[edit snmp v3]
root# edit usm local-engine user username1

[edit snmp v3 usm local-engine user username1]
```

```
root# set privacy-des privacy-password password

[edit snmp v3 usm local-engine user username1]
root# up

[edit snmp v3 usm local-engine]
root# set user username2 authentication-sha authentication-password
authentication-password

[edit snmp v3 usm local-engine]
root# edit user username2

[edit snmp v3 usm local-engine user username2]
root# set privacy-none

[edit snmp v3 usm local-engine user username2]
root# up

[edit snmp v3 usm local-engine]
root# set user username3 authentication-none privacy-none

[edit snmp v3 usm local-engine]
root# set user username4 authentication-md5 authentication-password
authentication-password

[edit snmp v3 usm local-engine]
root# edit user username4

[edit snmp v3 usm local-engine user username4]
root# set privacy-3des privacy-password password

[edit snmp v3 usm local-engine user username4]
root# up

[edit snmp v3 usm local-engine]
root# set user username5 authentication-sha authentication-password
authentication-password

[edit snmp v3 usm local-engine]
root# edit user username5

[edit snmp v3 usm local-engine user username5]
root# set privacy-aes128 privacy-password password

[edit snmp v3 usm local-engine user username5]
root# up

[edit snmp v3 usm local-engine]
root# up

[edit snmp v3 usm]
root# up

[edit snmp v3]
root# show
```

```
usm {
    local-engine {
      user username1 {
         authentication-md5 {
            authentication-key
            "$9$ZQGDk5T36Apf5reM8dVwYg4GDHkP3nC.mF/CtOBxN-VwgjHqQ36PfylKWx7P5TzCtI
            RSrKMIRbsY2aJCtpB1hrevL7-9AxNbs4ojHqf369ApB1hpuWLxNbw.Pf5n/1RhSevcSgoG
            DkqTz3/9pEcylvWFn/tOBEhgoaUDkTQnpu1AtORhyKvgoaJik"; ## SECRET-DATA
         }
         privacy-des {
            privacy-key
            "$9$kqmT6/tu1Rn68XNdg4aZUDqm5T3tpBzFAOBIcSwY24aUP5Q9tu3nWLx-ws36/CBIrl
            M8xNrloJZGiHBIRSyK8X7Vs2O1wYoJDjP5QntuO1RSyKRE-VwYoaz3n6pOylKMX7vMUjqm
            TQ/CtOORevWL7-ApOIcSeKUji.mT/9pREy1IclKWx7UjiHfT"; ## SECRET-DATA

         }
      }
      user username2 {
         authentication-sha {
            authentication-key
            "$9$B/bIylLX-2gJ-dgJGjq.O1Icevx7VgaZylK87NY2z3n/9pIEcKvLREeW8Xbwz3n/Cu
            1RhSevOOxN-VY2z3n9AO1RheM8RE-VY2aJ36/CAO1RhKvLyrZUHqf5QFn69pOOReM8IRhr
            Kv7Nk.mf5F9Ap1IcuOdbws4o69Cp1RleW-dsLXDikqQzlKvMNd"; ## SECRET-DATA
         }
             privacy-none;
      }
      user username3 {
         authentication-none;
         privacy-none;
      }
      user username4 {
         authentication-md5 {
            authentication-key
            "$9$dDVwgGUHqfTZGuOIEeKM8XxVw2gJHkPoai.P5F3SrlKMXs24DHqJZpOBRSyJGUjP56
            /tuBI6/vW8LN-P5T3nCuO1hylmfSrvWx7s24ZHqmfT3nCTQRhSrvMoJZGk.n/CtO1AtX7V
            wg4UjH.mT9ApO1Rik.5F39CX7NbwgUDkTQnf5F/CpB1X7N-Yg"; ## SECRET-DATA
         }
         privacy-3des {
            privacy-key
            "$9$bBwYoDjqmTzUDO1EcvM8X7NwY4oGq.5JZkP5Qn6reKM8724aiqmGUOBIhrlGDjH5Q9
            CuOIE9CWLXxdV5Qz6/tO1RSlKfTreWLN-24aUqmfTz6/tzFhSreW8JGUD.P/Ctu1Rpu7-w
            YoajHqPfzApOBRhk.PQn6At7-dsYoji.zF/TQnCtOIR7-dVgo"; ## SECRET-DATA
         }
      }
      user username5 {
         authentication-sha {
            authentication-key
            "$9$P5n/OORleWREeWL7Vbmf5F9AB1heM8n/Cu1IrloJZUDk5QFCAOTQ9puOcSoJZUjqfT
            z39A.mBIRhrloJZDi.fTz9tuTQRhrlMWJGUji.fTzCAOn68X-VY24aZGDk.mT9tu5Tz6CA
            1IdbwY2aDikf5Fq.EcSyKvGDjkfT/9pREyOOxNdV4o/CAtIE"; ## SECRET-DATA
         }
         privacy-aes128 {
            privacy-key
            "$9$wGYJGq.56/t5T/tuBEhbsYoUjmPQ/ApJGDkPfn6KMWLX-YgoDjq2gUHk.zFKMWLxds
```

```
                24aUjVbmf5Qn6KMWX7Vs24Uik2g5Qn6AtM8Lx7Vs24DjqJZpOIEyrevW8X-Vb2UikY24ZD
                jPfRhcyrvX7-sYodVTzF39C8Xx-s2GUH5T3q.O1REeKGDjifT"; ## SECRET-DATA
            }
        }
    }
}
```

13.4.8 Defining access privileges for an SNMP group

The SNMPv3 uses the VACM through which you can configure the access privileges granted to a group. Access is controlled by filtering the MIB objects that are available for a specific operation through a predefined view. You assign views to determine the objects that are visible for read, write, and notify operations for a particular group. You assign the view by using a particular context, a particular security model (*v1, v2c*, or *usm*), and particular security level (*authenticated, privacy*, or *none*). For information about how to configure views, see 13.3.11, “Configuring MIB views” on page 420.

You define user access to management information at the `[edit snmp v3 vacm]` hierarchy level. All access control within VACM operates on groups, which are collections of users as defined by USM, or community strings as defined in the SNMPv1 and SNMPv2c security models. The term *security name* refers to these generic users. The group to which a specific security name belongs is configured at the `[edit snmp v3 vacm security-to-group]` hierarchy level. That security name can be associated with a group defined at the `[edit snmp v3 vacm security-to-group]` hierarchy level. A group identifies a collection of SNMP users that share the same access policy.

You then define the access privileges that are associated with a group at the `[edit snmp v3 vacm access]` hierarchy level. Access privileges are defined by using views. For each group, you can apply different views depending on the SNMP operation. Examples include read (**get**, **getNext**, or **getBulk**), write (set), notifications, the security level used (authentication, privacy, or none), and the security model (v1, v2c, or usm) used within an SNMP request.

You configure members of a group with the **security-name** statement. For v3 packets using USM, the security name is the same as the user name. For SNMPv1 or SNMPv2c packets, the security name is determined based on the community string. Security names are specific to a security model. You might also be configuring VACM access policies for SNMPv1 or SNMPv2c packets. In this case, assign security names to groups for each security model SNMPv1 or SNMPv2c at the `[edit snmp v3 vacm security-to-group]` hierarchy level. You must also associate a security name with an SNMP community at the `[edit snmp v3 snmp-community community-index]` hierarchy level.

To configure the access privileges for an SNMP group, include statements at the `[edit snmp v3 vacm]` hierarchy level:

```
[edit snmp v3 vacm]
root# set access group group-name default-context-prefix security-model [any | usm
| v1 | v2c] security-level [authentication | none | privacy] notify-view view-name
read-view view-name write-view view-name
[edit snmp v3 vacm]
root# set security-to-group security-model [usm | v1 | v2c] security-name
security-name group group-name
```

13.4.9 Configuring the access privileges granted to a group

This topic includes the following sections:

- Configuring the group
- Configuring the security model
- Configuring the security level
- Associating MIB views with an SNMP user group

Configuring the group

To configure the access privileges that are granted to a group, include the **group** statement at the [edit snmp v3 vacm access] hierarchy level:

```
[edit snmp v3 vacm access]
root# set group group-name
```

where *group-name* is a collection of SNMP users that belong to a common SNMP list that defines an access policy. Users that belong to a particular SNMP group inherit all access privileges granted to that group.

Configuring the security model

To configure the security model, include the **security-model** statement at the [edit snmp v3 vacm access group *group-name* default-context-prefix] hierarchy level:

```
[edit snmp v3 vacm access group group-name default-context-prefix]
root# set security-model [any | usm | v1 | v2c]
```

where:

any	is any security model.
usm	is the SNMPv3 security model.
v1	is the SNMPV1 security model.
v2c	is the SNMPv2c security model.

Configuring the security level

To configure the access privileges granted to packets with a particular security level, include the **security-level** statement at the [edit snmp v3 vacm access group *group-name* default-context-prefix security-model [any | usm | v1 | v2c]] hierarchy level:

```
[edit snmp v3 vacm access group group-name default-context-prefix security-model
[any | usm | v1 | v2c]]
root# set security-level [authentication | none | privacy]
```

where:

none	provides no authentication and no encryption.
authentication	provides authentication but no encryption.
privacy	provides authentication and encryption.

Access privileges and security levels: Access privileges are granted to all packets with a security level equal to or greater than the level that is configured. If you are configuring the SNMPv1 or SNMPv2c security model, use *none* as your security level. If you are configuring the SNMPv3 security model, USM, use the *authentication*, *none*, or *privacy* security level.

Associating MIB views with an SNMP user group

MIB views define access privileges for members of a group. Separate views can be applied for each SNMP operation (read, write, and notify) within each security model (usm, v1, and v2c) and each security level (authentication, none, and privacy) that is supported by SNMP.

To associate MIB views with an SNMP user group, include the following statements at the `[edit snmp v3 vacm access group group-name default-context-prefix security-model [any | usm | v1 | v2c] security-level [authentication | none | privacy]]` hierarchy level:

```
[edit snmp v3 vacm access group group-name default-context-prefix security model
[any | usm | v1 | v2c] security-level [authentication | none | privacy]]
root# set notify-view view-name read-view view-name write-view view-name
```

> **Requirement:** You must associate at least one view (notify, read, or write) at the `[edit snmp v3 vacm access group group-name default-context-prefix security-model (any | usm | v1 | v2c) security-level (authentication | none | privacy)]` hierarchy level.
>
> You must configure the MIB view at the `[edit snmp view view-name]` hierarchy level. For information about how to configure MIB views, see 13.3.11, “Configuring MIB views” on page 420.

This topic includes the following sections related to this configuration:

- Configuring the notify view
- Configuring the read view
- Configuring the write view

Configuring the notify view

To associate notify access with an SNMP user group, include the **`notify-view`** statement at the `[edit snmp v3 vacm access group group-name default-context-prefix security-model [any | usm | v1 | v2c] security-level [authentication | none | privacy]]` hierarchy level:

```
[edit snmp v3 vacm access group group-name default-context-prefix security-model
[any | usm | v1 | v2c] security-level [authentication | none | privacy]]
root# set notify-view view-name
```

where *`view-name`* specifies the notify access, which is a list of notifications that can be sent to each user in an SNMP group. A view name cannot exceed 32 characters.

Configuring the read view

To associate a read view with an SNMP group, include the **`read-view`** statement at the `[edit snmp v3 vacm access group group-name default-context-prefix security-model [any | usm | v1 | v2c] security-level [authentication | none | privacy]]` hierarchy level:

```
[edit snmp v3 vacm access group group-name default-context-prefix security-model
[any | usm | v1 | v2c] security-level [authentication | none | privacy]]
root# set read-view view-name
```

where *`view-name`* specifies read access for an SNMP user group. A view name cannot exceed 32 characters.

Configuring the write view

To associate a write view with an SNMP user group, include the **write-view** statement at the `[edit snmp v3 vacm access group group-name default-context-prefix security-model [any | usm | v1 | v2c] security-level [authentication | none | privacy]]` hierarchy level:

```
[edit snmp v3 vacm access group group-name default-context-prefix security-model
[any | usm | v1 | v2c] security-level [authentication | none | privacy]]
root# set write-view view-name
```

where *view-name* specifies write access for an SNMP user group. A view name cannot exceed 32 characters.

13.4.10 Example: Defining access privileges

Example 13-5 shows how to define the access privileges.

Example 13-5 Defining access privileges

```
[edit snmp v3 vacm]
root# set access group group1 default-context-prefix security-model usm
security-level privacy notify-view nv1 read-view rv1 write-view wv1

[edit snmp v3 vacm]
root# set access group group2 default-context-prefix security-model usm
security-level authentication read-view rv2 write-view wv2

[edit snmp v3 vacm]
root# set access group group3 default-context-prefix security-model usm
security-level privacy read-view rv1 write-view wv1

[edit snmp v3 vacm]
root# show
access {
   group group1 {
      default-context-prefix {
         security-model usm {
             security-level privacy {
                 read-view rv1;
                 write-view wv1;
                 notify-view nv1;
             }
         }
      }
   }
   group group2 {
      default-context-prefix {
         security-model usm {
             security-level authentication {
                 read-view rv2;
                 write-view wv2;
             }
         }
      }
   }
   group group3 {
```

```
        default-context-prefix {
           security-model usm {
               security-level none {
                   read-view rv3;
                   write-view wv3;
               }
           }
        }
    }
}
```

13.4.11 Assigning security names to groups

To assign security names to groups, include the following statements at the [edit snmp v3 vacm security-to-group] hierarchy level:

```
[edit snmp v3 vacm security-to-group]
root# set security-model [usm | v1 | v2c] security-name security-name group
group-name
```

This topic includes the following sections:

- Configuring the security model
- Configuring the security name
- Configuring the group

Configuring the security model

To configure the security model, include the **security-model** statement at the [edit snmp v3 vacm security-to-group] hierarchy level:

```
[edit snmp v3 vacm security-to-group]
root# set security-model [usm | v1 | v2c]
```

where:

usm	is the SNMPv3 security model.
v1	is the SNMPv1 security model.
v2c	is the SNMPv2 security model.

Configuring the security name

To associate a security name with a user or community string, include the **security-name** statement at the [edit snmp v3 vacm security-to-group security-model [usm | v1 | v2c]] hierarchy level:

```
[edit snmp v3 vacm security-to-group security-model [usm | v1 | v2c]]
root# set security-name security-name
```

where *security-name* is the user name that is configured at the [edit snmp v3 usm local-engine user username] hierarchy level. For SNMPv1 and SNMPv2c, the security name is the community string configured at the [edit snmp v3 snmp-community community-index] hierarchy level. For information about configuring user names, see 13.4.4, “Creating SNMPv3 users” on page 428. For information about configuring a community string, see 13.4.21, “Configuring the SNMPv3 community” on page 450.

SNMPv1 and SNMPv2c security name: The USM security name is separate from the SNMPv1 and SNMPv2c security name. If you are supporting SNMPv1 and SNMPv2c, you must configure separate security names within the security-to-group configuration at the `[edit snmp v3 vacm access]` hierarchy level.

Configuring the group

After you create users, v1, or v2 security names, you associate them with a group. A *group* is a set of security names that belong to a particular security model. A group defines the access rights for all users who belong to it. Access rights define what SNMP objects can be read, written to, or created. A group also defines what notifications a user is allowed to receive.

If you already have a group that is configured with all of the view and access permissions that you want to give a user, you can add the user to that group. If you want to give a user view and access permissions that no other groups have, create a group and add the user to it. You can also create a group and add a user to it if you do not already have any groups configured.

To configure the access privileges that are granted to a group, include the **group** statement at the `[edit snmp v3 vacm security-to-group security-model [usm | v1 | v2c] security-name security-name]` hierarchy level:

```
[edit snmp v3 vacm security-to-group security-model [usm | v1 | v2c] security-name
security-name]
root# set group group-name
```

where *group-name* identifies a collection of SNMP security names that share the same access policy.

For more information about groups, see 13.4.8, “Defining access privileges for an SNMP group” on page 434.

Example 13-6 shows how to assign security names to groups.

Example 13-6 Configuring a security group

```
[edit snmp v3 vacm]
root# set security-to-group security-model usm security-name user1 group group1

[edit snmp v3 vacm]
root# set security-to-group security-model usm security-name user2 group group2

[edit snmp v3 vacm]
root# set security-to-group security-model usm security-name user3 group group3

[edit snmp v3 vacm]
root# show
security-to-group {
   security-model usm {
      security-name user1 {
         group group1;
      }
      security-name user2 {
         group group2;
      }
```

```
        security-name user3 {
            group group3;
        }
    }
}
```

13.4.12 Configuring SNMPv3 traps on a device running Junos software

In SNMPv3, traps and informs are created by configuring the `notify`, `target-address`, and `target-parameters` parameters. *Traps* are unconfirmed notifications, and *informs* are confirmed notifications. This section explains how to configure SNMP traps. For information about configuring SNMP informs, see 13.4.18, “Configuring SNMP informs” on page 447.

The target address defines the address of the management application and the parameters to use in sending notifications. Target parameters define the message processing and security parameters that are used in sending notifications to a particular management target. With SNMPv3, you can also define SNMPv1 and SNMPv2c traps.

Access privileges: When you configure SNMP traps, ensure that your configured access privileges allow the traps to be sent. Access privileges are configured at the `[edit snmp v3 vacm access]` and `[edit snmp v3 vacm security-to-group]` hierarchy levels.

To configure SNMP traps, include the following statements at the `[edit snmp v3]` hierarchy level:

```
[edit snmp v3]
root# set notify name tag tag-name type [trap | inform]

[edit snmp v3]
root# set notify-filter name oid object-identifier [include | exclude]
[edit snmp v3]
root# set target-address target-address-name address address address-mask
address-mask port port-number routing-instance instance tag-list tag-list
target-parameters target-parameters-name

[edit snmp v3]
root# set target-parameters target-parameters-name notify-filter profile-name
parameters message-processing-model [v1 | v2c | v3] security-model [usm | v1 |
v2c] security-level [authentication | none | privacy] security-name security-name
```

13.4.13 Configuring the SNMPv3 trap notification

The notify statement specifies the type of notification and contains a single tag. The tag defines a set of target addresses to receive a trap. The tag list contains one or more tags and is configured at the `[edit snmp v3 target-address target-address-name]` hierarchy level. If the tag list contains this tag, Junos software sends a notification to all the target addresses that are associated with this tag.

To configure the trap notifications, include the **`notify`** statement at the `[edit snmp v3]` hierarchy level:

```
[edit snmp v3]
root# set notify name tag tag-name type trap
```

where *name* is the name assigned to the notification, and *tag-name* defines the target addresses that are sent this notification. All the target-addresses that have this tag in their tag list are sent this notification. The *tag-name* is not included in the notification. The *trap* argument is the type of notification.

> **Notify entry name:** Each notify entry name must be unique. Junos software supports two types of notification: trap and inform.

For information about how to configure the tag list, see “Configuring the tag list” on page 443.

Example 13-7 shows how to specify three sets of destinations to send traps.

Example 13-7 Configuring SNMPv3 trap notification

```
[edit snmp v3]
root# set notify n1 tag router1 type trap

[edit snmp v3]
root# set notify n2 tag router2 type trap

[edit snmp v3]
root# set notify n3 tag router3 type trap

[edit snmp v3]
root# show
notify n1 {
   type trap;
   tag router1;
}
notify n2 {
   type trap;
   tag router2;
}
notify n3 {
   type trap;
   tag router3;
}
```

13.4.14 Configuring the trap notification filter

SNMPv3 uses the notify filter to define the traps or objects from where the traps are sent to the NMS. The trap notification filter limits the type of traps that are sent to the NMS.

Each OID represents a subtree of the MIB object hierarchy. The subtree can be represented either by a sequence of dotted integers, such as 1.3.6.1.2.1.2, or by its subtree name, such as *interfaces*. You can also use the wildcard character asterisk (*) in the OID to specify OIDs that match a particular pattern.

To configure the trap notifications filter, include the **notify-filter** statement at the `[edit snmp v3]` hierarchy level:

```
[edit snmp v3]
root# set notify-filter profile-name
```

where *profile-name* is the name assigned to the notify filter.

By default, the OID is set to **include**. To define access to traps or objects from traps, include the **oid** statement at the [edit snmp v3 notify-filter *profile-name*] hierarchy level:

```
[edit snmp v3 notify-filter profile-name]
root# set oid oid [include | exclude]
```

where:

oid	is the object identifier. All MIB objects represented by this statement have the specified OID as a prefix. It can be specified either by a sequence of dotted integers or by a subtree name.
include	includes the subtree of MIB objects represented by the specified OID.
exclude	excludes the subtree of MIB objects represented by the specified OID.

13.4.15 Configuring the trap target address

The target address defines the address of the management application and parameters that are used in sending notifications. It can also identify management stations that are allowed to use specific community strings. When you receive a packet with a recognized community string and a tag is associated with it, Junos software looks up all the target addresses with this tag. The software then verifies that the source address of this packet matches one of the configured target addresses.

Address mask: You must configure the address mask when you configure the SNMP community.

To specify where you want the traps to be sent and define which SNMPv1 and SNMP2vc packets are allowed, include the **target-address** statement at the [edit snmp v3] hierarchy level:

```
[edit snmp v3]
root# set target-address target-address-name
```

where *target-address-name* is the string that identifies the target address.

To configure the target address properties, include the following statements at the [edit snmp v3 target-address *target-address-name*] hierarchy level:

```
[edit snmp v3 target-address target-address-name]
root# set address address address-mask address-mask port port-number
routing-instance instance tag-list tag-list target-parameters
target-parameters-name
```

This section includes the following topics:

- Configuring the address
- Configuring the address mask
- Configuring the port
- Configuring the routing instance
- Configuring the tag list
- Applying target parameters

Configuring the address

To configure the address, include the **address** statement at the [`edit snmp v3 target-address` *`target-address-name`*] hierarchy level:

```
[edit snmp v3 target-address target-address-name]
root# set address address
```

where *`address`* is the SNMP target address.

Configuring the address mask

The address mask specifies a set of addresses that are allowed to use a community string and verifies the source addresses for a group of target addresses. To configure the address mask, include the **address-mask** statement at the [`edit snmp v3 target-address` *`target-address-name`*] hierarchy level:

```
[edit snmp v3 target-address target-address-name]
root# address-mask address-mask
```

where *`address-mask`* combined with the address defines a range of addresses.

For information about how to configure the community string, see 13.4.21, “Configuring the SNMPv3 community” on page 450.

Configuring the port

By default, the UDP port is set to 162. To configure a different port number, include the *`port`* statement at the [`edit snmp v3 target-address` *`target-address-name`*] hierarchy level:

```
[edit snmp v3 target-address target-address-name]
root# set port port-number
```

where *`port-number`* is the SNMP target port number.

Configuring the routing instance

Traps are sent over the default routing instance. To configure the routing instance for sending traps, include the **routing-instance** statement at the [`edit snmp v3 target-address` *`target-address-name`*] hierarchy level:

```
[edit snmp v3 target-address target-address-name]
root# set routing-instance instance
```

where *`instance`* is the name of the routing instance.

To configure a routing instance within a logical system, specify the logical system name followed by the routing instance name. Use a forward slash (/) to separate the two names, for example, `test-lr/test-ri`. To configure the default routing instance on a logical system, specify the logical system name followed by default, for example, `test-lr/default`.

Configuring the tag list

Each **target-address** statement can have one or more tags configured in its tag list. Each tag can be displayed in more than one tag list. When a significant event occurs on the network device, the tag list identifies the targets to which a notification is sent. To configure the tag list, include the **tag-list** statement at the [`edit snmp v3 target-address` *`target-address-name`*] hierarchy level:

```
[edit snmp v3 target-address target-address-name]
root# set tag-list tag-list
```

where *tag-list* specifies one or more tags as a space-separated list enclosed within double quotation marks. For an example of tag list configuration, see Example 13-8.

> **Allowing for the sending of traps in access privileges:** When you configure SNMP traps, make sure that your configured access privileges allow the traps to be sent. Configure access privileges at the [`edit snmp v3 vacm access`] hierarchy level.

Applying target parameters

The **target-parameters** statement at the [`edit snmp v3`] hierarchy level applies the target parameters that are configured at the [`edit snmp v3 target-parameters` *target-parameters-name*] hierarchy level.

To reference configured target parameters, include the **target-parameters** statement at the [`edit snmp v3 target-address` *target-address-name*] hierarchy level:

```
[edit snmp v3 target-address target-address-name]
root# set target-parameters target-parameters-name
```

where *target-parameters-name* is the name associated with the message processing and security parameters that are used in sending notifications to a particular management target.

13.4.16 Example: Configuring the tag list

In Example 13-8, two tag entries, `router1` and `router2`, are defined at the [`edit snmp v3 notify` *notify-name*] hierarchy level. When an event triggers a notification, Junos software sends a trap to all target addresses that have `router1` or `router2` configured in their target-address tag list. The result is in the first two targets getting one trap each and the third target getting two traps.

Example 13-8 Configuring the tag list

```
[edit snmp v3]
root# set notify n1 tag router1 type trap

[edit snmp v3]
root# set notify n2 tag router2 type trap

[edit snmp v3]
root# set target-address ta1 address 10.1.1.1 address-mask 255.255.255.0 port 162
tag-list router1 target-parameters tp1

[edit snmp v3]
root# set target-address ta2 address 10.1.1.2 address-mask 255.255.255.0 port 162
tag-list router2 target-parameters tp2

[edit snmp v3]
root# set target-address ta3 address 10.1.1.3 address-mask 255.255.255.0 port 162
tag-list "router1 router2" target-parameters tp3

[edit snmp v3]
root# show
target-address ta1 {
   address 10.1.1.1;
   port 162;
   tag-list router1;
```

```
   address-mask 255.255.255.0;
   target-parameters tp1;
}
target-address ta2 {
   address 10.1.1.3;
   port 162;
   tag-list router2;
   address-mask 255.255.255.0;
   target-parameters tp2;
}
target-address ta3 {
   address 10.1.1.3;
   port 162;
   tag-list "router1 router2";
   address-mask 255.255.255.0;
   target-parameters tp3;
}
```

13.4.17 Defining and configuring the trap target parameters

Target parameters define the message processing and security parameters that are used in sending notifications to a particular management target.

To define a set of target parameters, include the **target-parameters** statement at the [edit snmp v3] hierarchy level:

```
[edit snmp v3]
root# set target-parameters target-parameters-name
```

where *target-parameters-name* is the name assigned to the target parameters.

To configure target parameter properties, include the following statements at the [edit snmp v3 target-parameters *target-parameter-name*] hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name]
root# notify-filter profile-name parameters message-processing-model [v1 | v2c |
V3] security-level [authentication | none | privacy] security-model [usm | v1 |
v2c] security-name security-name
```

This topic includes the following sections:

- Applying the trap notification filter
- Configuring the target parameters

Applying the trap notification filter

To apply the trap notification filter, include the **notify-filter** statement at the [edit snmp v3 target-parameters *target-parameter-name*] hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name]
root# set notify-filter profile-name;
```

where *profile-name* is the name of a configured notify filter. For information about configuring notify filters, see 13.4.14, “Configuring the trap notification filter” on page 441.

Configuring the target parameters

To configure target parameter properties, include the following statements at the [`edit snmp v3 target-parameters` *`target-parameter-name`* `parameters`] hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name parameters]
root# set message-processing-model [v1 | v2c | v3]
root# set security-model [usm | v1 | v2c]
root# set security-level [authentication | none | privacy]
root# set security-name security-name
```

This section includes the following topics:

- Configuring the message processing model
- Configuring the security model
- Configuring the security level
- Configuring the security name

Configuring the message processing model

The message processing model defines which version of SNMP to use when generating SNMP notifications. To configure the message processing model, include the **`message-processing-model`** statement at the [`edit snmp v3 target-parameters` *`target-parameter-name`* `parameters`] hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name parameters]
root# set message-processing-model [v1 | v2c | v3]
```

where:

`v1`	is the SNMPv1 message processing model.
`v2c`	is the SNMPv2c message processing model.
`v3`	is the SNMPV3 message processing model.

Configuring the security model

To define the security model to use when generating SNMP notifications, include the **`security-model`** statement at the [`edit snmp v3 target-parameters` *`target-parameter-name`* `parameters`] hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name parameters]
root# set security-model [usm | v1 | v2c]
```

where:

`usm`	is the SNMPv3 security model.
`v1`	is the SNMPv1 security model.
`v2c`	is the SNMPv2c security model.

Configuring the security level

The security-level statement specifies whether the trap is authenticated and encrypted before it is sent. To configure the security level to use when generating SNMP notifications, include the **`security-level`** statement at the [`edit snmp v3 target-parameters` *`target-parameter-name`* `parameters`] hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name parameters]
root# set security-level [authentication | none | privacy]
```

where:

`authentication`	provides authentication but no encryption.
`none`	means no security. It provides no authentication and no encryption.
`privacy`	provides authentication and encryption.

SNMPv1 or SNMPv2c configuration: If you are configuring the SNMPv1 or SNMPv2c security model, use `none` as your security level. If you are configuring the SNMPv3 (USM) security model, use the `authentication` or `privacy` security level.

Configuring the security name

To configure the security name to use when generating SNMP notifications, include the **`security-name`** statement at the [`edit snmp v3 target-parameters` *`target-parameter-name`* `parameters`] hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name parameters]
root# set security-name security-name
```

If the USM security model is used, **`security-name`** identifies the user that is used when the notification is generated. If the v1 or v2c security models are used, **`security-name`** identifies the SNMP community that is used when the notification is generated.

Access privileges for the group: The access privileges for the group associated with a security name must allow this notification to be sent.

If you are using the v1 or v2 security models, the security name at the [`edit snmp v3 vacm security-to-group`] hierarchy level must match the security name at the [`edit snmp v3 snmp-community` *`community-index`*] hierarchy level.

13.4.18 Configuring SNMP informs

Junos software supports two types of notifications: traps and informs. With traps, the receiver does not send any acknowledgement when it receives a trap. Therefore, the sender cannot determine if the trap was received. A trap might be lost because a problem occurred during transmission. To increase reliability, an inform is similar to a trap except that the inform is stored and retransmitted at regular intervals until one of the following conditions occurs:

- The receiver (target) of the inform returns an acknowledgment to the SNMP agent.
- A specified number of unsuccessful retransmissions has been attempted and the agent discards the inform message.

If the sender never receives a response, the inform can be sent again. Thus, informs are more likely to reach their intended destination than traps are. Informs use the same communications channel as traps (same socket and port) but have different protocol data unit (PDU) types.

Informs are more reliable than traps, but they use more network and router resources. Unlike a trap, an inform is held in memory until a response is received or the timeout is reached. Also, traps are sent only once. An inform might be retried several times. Use informs when it is important for the SNMP manager to receive all notifications. However, if you are more concerned about network traffic or router memory, use traps.

Figure 13-1 shows the SNMP inform request and response.

Figure 13-1 SNMP inform request and response

For information about configuring SNMP traps, see 13.4.12, “Configuring SNMPv3 traps on a device running Junos software” on page 440.

13.4.19 Configuring the remote engine and remote user

To send inform messages to an SNMPv3 user on a remote device, you must first specify the engine identifier for the SNMP agent on the remote device where the user resides. The remote engine ID is used to compute the security digest for authenticating and encrypting packets sent to a user on the remote host. When sending an inform message, the agent uses the credentials of the user that is configured on the remote engine, which is the inform target.

To configure a remote engine and remote user to receive and respond to SNMP informs, include the following statements at the `[edit snmp v3]` hierarchy level:

```
[edit snmp v3]
root# set usm remote-engine engine-id user username [authentication-md5
authentication-key key] [authentication-none] [authentication-sha authentication-key
key] [privacy-3des privacy-key key] [privacy-aes128 privacy-key key] [privacy-des
privacy-key key] [privacy-none]
```

where for informs:

remote-engine *engine-id*
: is the identifier for the SNMP agent on the remote device where the user resides.

user *username*
: is the user on a remote SNMP engine who receives the informs. The informs that are generated can be *unauthenticated*, *authenticated*, or *authenticated_and_encrypted*, depending on the security level of the SNMPv3 user that is configured on the remote engine, which is the inform receiver. The authentication key is used for generating MAC

code. The privacy key is used to encrypt the inform PDU part of the message.

13.4.20 Configuring the inform notification type and target address

To configure the inform notification type and target information, include the following statements at the [edit snmp v3] hierarchy level:

```
[edit snmp v3]
root# set notify name tag tag-name type [trap | inform]
[edit snmp v3]
root# set target-address target-address-name address address address-mask
address-mask inform-retry-count number inform-timeout seconds port port-number
routing-instance instance tag-list tag-list target-parameters
target-parameters-name
[edit snmp v3]
root# set target-parameters target-parameters-name notify-filter profile-name
parameters message-processing-model [v1 | v2c | v3] security-model [usm | v1 |
v2c] security-level [authentication | none | privacy] security-name security-name
```

where:

notify *name* — is the name assigned to the notification. Each notify entry name must be unique.

tag *tag-name* — defines the target addresses that are sent this notification. The notification is sent to all target addresses that have this tag in their tag list. The *tag-name* is not included in the notification. For information about how to configure the tag list, see “Configuring the tag list” on page 443.

type inform — is the type of notification.

target-address *target-address-name* — identifies the target address. The target address defines the address of the management application and parameters that are used to respond to informs.

inform-timeout *seconds* — is the number of seconds to wait for an acknowledgment. If no acknowledgment is received within the timeout period, the inform is retransmitted. The default timeout is 15 seconds.

inform-retry-count *number* — is the maximum number of times an inform is transmitted if no acknowledgment is received. The default is 3. If no acknowledgment is received after the inform is transmitted the maximum number of times, the inform message is discarded.

message-processing-model — defines which version of SNMP to use when SNMP notifications are generated. Informs require a v3 message processing model.

security-model — defines the security model to use when SNMP notifications are generated. Informs require a **usm** security model.

security-level specifies whether the inform is authenticated and encrypted before it is sent. For the **usm** security model, the security level must be one of the following:

- authentication, which provides authentication but no encryption.
- privacy, which provides authentication and encryption.

security-name identifies the user name that is used when generating the inform.

13.4.21 Configuring the SNMPv3 community

The SNMP community defines the relationship between an SNMP server system and the client systems. This statement is optional.

To configure the SNMP community, include the **snmp-community** statement at the [edit snmp v3] hierarchy level:

```
[edit snmp v3]
root# set snmp-community community-index
```

where *community-index* is the index for the SNMP community.

To configure the SNMP community properties, include the following statements at the [edit snmp v3 snmp-community *community-index*] hierarchy level:

```
[edit snmp v3 snmp-community community-index]
root# set community-name community-name
root# set security-name security-name
root# set tag tag-name
```

This section includes the following topics:

- Configuring the community name
- Configuring the security names
- Configuring the tag

Configuring the community name

The community name defines the SNMP community. The SNMP community authorizes SNMPv1 or SNMPv2c clients. The access privileges that are associated with the configured security name define which MIB objects are available and the operations (read, write, or notify) that are allowed on those objects.

To configure the SNMP community name, include the **community-name** statement at the [edit snmp v3 snmp-community *community-index*] hierarchy level:

```
[edit snmp v3 snmp-community community-index]
root# set community-name community-name
```

where *community-name* is the community string for an SNMPv1 or SNMPv2c community. If it is unconfigured, it is the same as the community index. If the community name contains spaces, enclose it in quotation marks (" ").

Unique community names: Community names must be unique. You cannot configure the same community name at the [`edit snmp community`] and [`edit snmp v3 snmp-community community-index`] hierarchy levels. The configured community name at the [`edit snmp v3 snmp-community community-index`] hierarchy level is encrypted.

You cannot view the community name after you configure it and commit your changes. In the CLI, the community name is concealed.

Configuring the security names

To assign a community string to a security name, include the **`security-name`** statement at the [`edit snmp v3 snmp-community community-index`] hierarchy level:

```
[edit snmp v3 snmp-community community-index]
root# set security-name security-name
```

where *`security-name`* is used when access control is set up.

The security-to-group configuration at the [`edit snmp v3 vacm`] hierarchy level identifies the group.

Matching security name: This security name must match the security name configured at the [`edit snmp v3 target-parameters target-parameters-name parameters`] hierarchy level when you configure traps.

Configuring the tag

To configure the tag, include the **`tag`** statement at the [`edit snmp v3 snmp-community community-index`] hierarchy level:

```
[edit snmp v3 snmp-community community-index]
root# set tag tag-name
```

where *`tag-name`* identifies the address of managers that are allowed to use a community string.

13.5 MIB support in Junos software

This topic includes the following sections:

- Standard SNMP MIBs supported by Junos software
- Loading MIB files to a network management station

13.5.1 Standard SNMP MIBs supported by Junos software

Table 13-3 on page 452 contains the list of standard SNMP MIBs and RFCs that are supported on various devices running Junos software. You can find the RFCs at the following website:

http://www.ietf.org

For a detailed interpretation of Juniper Networks enterprise-specific MIBs, see “Juniper Networks Enterprise-Specific MIBs” in *JUNOS Software Network Management Configuration Guide*, GA32-0698.

M, E, and S columns: In Table 13-3, a value of 1 in any of the platform columns (M, E, or S) denotes that the corresponding MIB is supported on that particular platform. A value of 0 denotes that the MIB is not supported on the platform.

Table 13-3 SNMP MIBs and RFCs

MIB or RFC	M	E	S
IEEE 802.1ab, *Link Layer Discovery Protocol (LLDP) MIB*	0	1	0
IEEE, 802.3ad, *Aggregation of Multiple Link Segments* The following tables and objects are supported: ► `dot3adAggPortTable`, `dot3adAggPortListTable`, `dot3adAggTable`, and `dot3adAggPortStatsTable` ► `dot3adAggPortDebugTable` only `dot3adAggPortDebugRxState`, `dot3adAggPortDebugMuxState`, `dot3adAggPortDebugActorSyncTransitionCount`, `dot3adAggPortDebugPartnerSyncTransitionCount`, `dot3adAggPortDebugActorChangeCount`, and `dot3adAggPortDebugPartnerChangeCount` ► `dot3adTablesLastChanged` **Gigabit Ethernet interfaces:** Gigabit Ethernet interfaces on j-series services routers do not support the 802.3ad MIB.	1	1	0
RFC 1155, Structure and Identification of Management Information for TCP/IP-based Internets	1	1	0
RFC 1157, A Simple Network Management Protocol (SNMP)	1	1	0
RFC 1195 Use of OSI IS-IS for Routing in TCP/IP and Dual Environments Only the `isisSystem`, `isisMANAreaAddr`, `isisAreaAddr`, `isisSysProtSupp`, `isisSummAddr`, `isisCirc`, `isisCircLevel`, `isisPacketCount`, `isisISAdj`, `isisISAdjAreaAddr`, `isisAdjIPAddr`, `isisISAdjProtSupp`, `isisRa`, and `isisIPRA` objects are supported.	1	0	0
RFC 1212, Concise MIB Definitions	1	1	0
RFC 1213, Management Information Base for Network Management of TCP/IP-Based Internets: MIB-II Junos software supports the following areas: ► MIB II and its SNMP version 2 derivatives, including: -- Statistics counters -- IP, except for `ipRouteTable`, which has been replaced by `ipCidrRouteTable` RFC 2096, IP Forwarding Table MIB -- SNMP management -- Interface management ► SNMPv1 **`Get`**, **`GetNext`** requests, and version 2 **`GetBulk`** request ► Junos software-specific secured access list ► Master configuration keywords ► Reconfigurations upon SIGHUP	1	1	0
RFC 1215, A Convention for Defining Traps for use with the SNMP Only MIB II SNMP version 1 traps and version 2 notifications are supported	1	1	0
RFC 1657, Definitions of Managed Objects for the Fourth Version of the Border Gateway Protocol (BGP-4) using SMIv2	1	1	0
RFC 1850, OSPF Version 2 Management Information Base Except for the `ospfOriginateNewLsas` and `ospfRxNewLsas` objects, the host table, and the traps `ospfOriginateLSA`, `ospfLsdbOverflow`, and `ospfLsdbApproachingOverflow`	1	1	1
RFC 1901, Introduction to Community-based SNMPv2	1	0	0
RFC 2011, SNMPv2 Management Information Base for the Internet Protocol Using SMIv2	1	1	1

MIB or RFC	M	E	S
RFC 2012, SNMPv2 Management Information Base for the Transmission Control Protocol Using SMIv2	1	1	1
RFC 2013, SNMPv2 Management Information Base for the User Datagram Protocol Using SMIv2	1	1	1
RFC 2024, Definitions of Managed Objects for Data Link Switching Using SMIv2 Except for the `dlswInterface` and `dlswSdlc` object groups; the `dlswDirLocateMacTable`, `dlswDirNBTable`, and `dlswDirLocateNBTable` tables; the `dlswCircuitDiscReasonLocal` and `dlswCircuitDiscReasonRemote` tabular objects; and the `dlswDirMacCacheNextIndex` and `dlswDirNBCacheNextIndex` scalar objects; read-only access.	1	0	0
RFC 2096, IP Forwarding Table MIB `ipCidrRouteTable` has been extended to include the tunnel name when the next hop is through an RSVP-signaled LSP.	1	1	0
RFC 2115, Management Information Base for Frame Relay DTEs Using SMIv2 `frDlcmiTable` only; `frCircuitTable` and `frErrTable` are not supported.	1	0	0
RFC 2233, The Interfaces Group MIB Using SMIv2 **Replacement of RFC 2233:** RFC 2233 has been replaced by RFC 2863 if MIB. However, Junos software supports both RFC 2233 and RFC 2863.	1	1	1
RFC 2287, Definitions of System-Level Managed Objects for Applications Applies to only the `sysApplInstallPkgTable`, `sysApplInstallElmtTable`, `sysApplElmtRunTable`, and `sysApplMapTable` objects.	1	1	1
RFC 2465, Management Information Base for IP Version 6: Textual Conventions and General Group Except for IPv6 interface statistics	1	0	0
RFC 2570, Introduction to Version 3 of the Internet-standard Network Management Framework	1	1	0
RFC 2571, An Architecture for Describing SNMP Management Frameworks Read-only access **Replacement of RFC 2571:** RFC 2571 has been replaced by RFC 3411. However, Junos software supports both RFC 2571 and RFC 3411.	1	1	1
RFC 2572, Message Processing and Dispatching for the Simple Network Management Protocol (SNMP) read-only access. **Replacement of RFC 2572:** RFC 2572 has been replaced by RFC 3412. However, Junos software supports both RFC 2572 and RFC 3412.	1	1	1
RFC 2576, Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework **Replacement of RFC 2576:** RFC 2576 has been replaced by RFC 3584. However, Junos software supports both RFC 2576 and RFC 3584.	1	1	1
RFC 2578, Structure of Management Information Version 2 (SMIv2)	1	1	0
RFC 2579, Textual Conventions for SMIv2	1	1	0
RFC 2580, Conformance Statements for SMIv2	1	1	0
RFC 2662, Definitions of Managed Objects for ADSL Lines	1	0	0
RFC 2665, Definitions of Managed Objects for the Ethernet-like Interface Types	1	1	1
RFC 2787, Definitions of Managed Objects for the Virtual Router Redundancy Protocol Except row creation, the `Set` operation, and the `vrrpStatsPacketLengthErrors` object.	1	1	1

MIB or RFC	M	E	S
RFC 2790, Host Resources MIB ► Only the `hrStorageTable`. The file systems `/`, `/config`, `/var`, and `/tmp` always return the same index number. When SNMP restarts, the index numbers for the remaining file systems might change. ► Only the objects of the `hrSystem` and `hrSWInstalled` groups.	1	1	1
RFC 2819, Remote Network Monitoring Management Information Base The `etherStatsTable` for Ethernet interfaces only and the objects `alarmTable`, `eventTable`, and `logTable`.	1	1	1
RFC 2863, The Interfaces Group MIB **Replacement of RFC 2233:** RFC 2863 replaces RFC 2233. However, Junos software supports both RFC 2233 and RFC 2863.	1	1	0
RFC 2864, The Inverted Stack Table Extension to the Interfaces Group MIB	1	0	0
RFC 2922, The Physical Topology (PTOPO) MIB	0	1	1
RFC 2925, Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations Only the objects `pingCtlTable`, `pingResultsTable`, `pingProbeHistoryTable`, `pingMaxConcurrentRequests`, `traceRouteCtlTable`, `traceRouteResultsTable`, `traceRouteProbeHistoryTable`, and `traceRouteHopsTable`.	1	1	1
RFC 2932, IPv4 Multicast Routing MIB	1	1	1
RFC 2933, Internet Group Management Protocol (IGMP) MIB	1	1	1
RFC 2934, Protocol Independent Multicast MIB for IPv4	1	1	1
RFC 2981, Event MIB	1	0	0
RFC 3014, Notification Log MIB	1	0	0
RFC 3019, IP Version 6 Management Information Base for The Multicast Listener Discovery Protocol	1	0	0
RFC 3410 Introduction and Applicability Statements for Internet-Standard Management Framework	1	1	0
RFC 3411, An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks **Replacement of RFC 2571:** RFC 3411 replaces RFC 2571. However, Junos software supports both RFC 3411 and RFC 2571.	1	1	0
RFC 3412, Message Processing and Dispatching for the Simple Network Management Protocol (SNMP) **Replacement of RFC 2572:** RFC 3412 replaces RFC 2572. However, Junos software supports both RFC 3412 and RFC 2572.	1	1	0
RFC 3413, Simple Network Management Protocol (SNMP) Applications Except for the proxy MIB.	1	1	1
RFC 3414, User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)	1	1	0
RFC 3415, View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)	1	1	0
RFC 3416, Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP) **Replacement of RFC 1905:** RFC 3416 replaces RFC 1905, which was supported in earlier versions of Junos software.	1	1	0
RFC 3417, Transport Mappings for the Simple Network Management Protocol (SNMP)	1	1	1

MIB or RFC	M	E	S
RFC 3418, Management Information Base (MIB) for the Simple Network Management Protocol (SNMP) **Replacement of RFC 1907:** RFC 3418 replaces RFC 1907, which was supported in earlier versions of Junos software.	1	1	0
RFC 3584, Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework	1	1	0
RFC 3592, Definitions of Managed Objects for the Synchronous Optical Network/Synchronous Digital Hierarchy (SONET/SDH) Interface Type	1	0	0
RFC 3621, Power Ethernet MIB	0	1	0
RFC 3637, Definitions of Managed Objects for the Ethernet WAN Interface Sublayer Except `etherWisDeviceTable`, `etherWisSectionCurrentTable`, and `etherWisFarEndPathCurrentTable`.	1	0	0
RFC 3811, Definitions of Textual Conventions (TCs) for Multiprotocol Label Switching (MPLS) Management	1	0	0
RFC 3812, Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Management Information Base (MIB) Read only access ► MPLS tunnels as interfaces are not supported. ► The following objects in the `TunnelResource` table are not supported: `mplsTunnelResourceMeanRate`, `mplsTunnelResourceMaxBurstSize`, `mplsTunnelResourceMeanBurstSize`, `mplsTunnelResourceExBurstSize`, `mplsTunnelResourceWeight`. ► `mplsTunnelPerfTable` and `mplsTunnelCRLDPResTable` are not supported. ► `mplsTunnelCHopTable` supported on ingress routers only. **Branch conflict:** The branch used by the proprietary LDP MIB, `ldpmib.mib`, conflicts with RFC 3812. The `ldpmib.mib` branch has been deprecated and replaced by `jnx-mpls-ldp.mib`.	1	0	0
RFC 3813, Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base (MIB) Read only access. `mplsInterfacePerfTable`, `mplsInSegmentPerfTable`, `mplsOutSegmentPerfTable`, `mplsInSegmentMapTable`, `mplsXCUp`, and `mplsXCDown` are not supported.	1	0	0
RFC 3815, Definitions of Managed Objects for the Multiprotocol Label Switching (MPLS), Label Distribution Protocol (LDP) Only `mplsLdpLsrID` and `mplsLdpSesPeerAddrTable`.	1	0	0
RFC 3826, The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model	1	1	0
RFC 4188, Definitions of Managed Objects for Bridges Supports 802.1D STP(1998). Supports only the following subtrees and objects: ► The `dot1dStp` subtree is supported on the IBM J series M modelEthernet Services Routers. ► The `dot1dTpFdbAddress`, `dot1dTpFdbPort`, and `dot1dTpFdbStatus` objects from the `dot1dTpFdbTable` of the `dot1dTp` subtree are supported on IBM J-series E model Ethernet switches.	1	1	0
RFC 4318, Definitions of Managed Objects for Bridges with Rapid Spanning Tree Protocol Supports 802.1w and 802.1t extensions for RSTP.	1	1	0
RFC 4363b, Q-Bridge VLAN MIB	1	1	0
RFC 4801, Definitions of Textual Conventions for Generalized Multiprotocol Label Switching (GMPLS) Management Information Base (MIB) Read-only access.	1	0	0

MIB or RFC	M	E	S
RFC 4802, Generalized Multiprotocol Label Switching (GMPLS) Traffic Engineering (TE) Management Information (MIB) Read-only access. `gmplsTunnelReversePerfTable`, `gmplsTeScalars`, `gmplsTunnelTable`, `gmplsTunnelARHopTable`, `gmplsTunnelCHopTable`, and `gmplsTunnelErrorTable` are not supported.	1	0	0
RFC 4803, Generalized Multiprotocol Label Switching (GMPLS) Label Switching Router (LSR) Management Information Base (MIB) Read-only access. `gmplsLabelTable` and `gmplsOutsegmentTable` are not supported. **GMPLS TE tables:** The tables in GMPLS TE (RFC 4802) and LSR (RFC 4803) MIBs are extensions of the corresponding tables from the MPLS TE (RFC 3812) and LSR (RFC 3813) MIBs and use the same index as the MPLS MIB tables.	1	0	0
Internet Assigned Numbers Authority, *IANAiftype Textual Convention MIB,* referenced by RFC 2233, available at: http://www.iana.org/assignments/ianaiftype-mib	1	1	1
Internet draft `draft-ietf-atommib-sonetaps-mib-10.txt`, *Definitions of Managed Objects for SONET Linear APS Architectures*, as defined under the Juniper Networks enterprise branch `[jnxExperiment]` only.	1	0	0
Internet draft `draft-ieft-bfd-mib-02.txt`, *Bidirectional Forwarding Detection Management Information Base* Represented by `mib-jnx-bfd-exp.txt` and implemented under the Juniper Networks enterprise branch `[jnxExperiment]`. Read only. Includes `bfdSessUp` and `bfdSessDown` traps. Does not support `bfdSessPerfTable` and `bfdSessMapTable`.	1	1	0
Internet draft `draft-ietf-idmr-igmp-mib-13.txt`, *Internet Group Management Protocol (IGMP) MIB*	1	1	0
Internet draft `draft-ietf-idr-bgp4-mibv2-04.txt`, *Definitions of Managed Objects for the Fourth Version of Border Gateway Protocol (BGP-4), Second Version* Only `jnxBgpM2PrefixInPrefixes`, `jnxBgpM2PrefixInPrefixesAccepted`, and `jnxBgpM2PrefixInPrefixesRejected` objects.	1	1	0
Internet draft `draft-reeder-snmpv3-usm-3desede-00.txt`, *Extension to the User-Based Security Model (USM) to Support Triple-DES EDE in 'Outside' CBC Mode*	1	1	0
Internet draft `draft-ietf-isis-wg-mib-07.txt`, *Management Information Base for IS-IS* Only `isisISAdjTable`, `isisISAdjAreaAddrTable`, `isisISAdjIPAddrTable`, and `isisISAdjProtSuppTable`.	1	1	1
Internet draft `draft-ietf-ppvpn-mpls-vpn-mib-04.txt`, *MPLS/BGP Virtual Private Network Management Information Base Using SMIv2* Only `mplsVpnScalars`, `mplsVpnVrfTable`, `mplsVpnPerTable`, and `mplsVpnVrfRouteTargetTable`.	1	0	0
Internet draft `draft-ietf-ospf-ospfv3-mib-11.txt`, *Management Information Base for OSPFv3* Represented by `mib-jnx-ospfv3mib.txt` and implemented under the Juniper Networks enterprise branch `{jnxExperiment}`. Support for `ospfv3NbrTable` only. Read only. Object names are prefixed by *jnx*. For example, `jnxOspfv3NbrTable`, `jnxOspfv3NbrAddressType`, and `jnxOspfv3NbrPriority`.	1	0	0
Internet draft `draft-ietf-idmr-pim-mib-09.txt`, *Protocol Independent Multicast (PIM) MIB*	1	1	0
ESO Consortium MIB, which is available at: http://www.snmp.com/eso/ **Replacement:** ESO Consortium MIB has been replaced by RFC 3826.	1	1	1

13.5.2 Loading MIB files to a network management station

For your NMS to identify and understand the MIB objects used by Junos software, you must first load the MIB files to your NMS using an MIB compiler. An MIB compiler is a utility that parses the MIB information, such as the MIB object name, IDs, and data type for the NMS.

You can download the Junos MIB package from the Enterprise-Specific MIBs and Traps section of the Junos Software Technical Publications index page at the following address:

http://www.juniper.net/techpubs/software/junos

The Junos MIB package is available in ZIP and TAR packages. You can download the appropriate format based on your requirements.

The Junos MIB package contains two folders: `StandardMibs` and `JuniperMibs`. The `StandardMibs` folder contains the standard MIBs and RFCs that are supported on devices running Junos software. The `JuniperMibs` folder contains the Juniper Networks enterprise-specific MIBs.

To load MIB files that are required for managing and monitoring devices running Junos software, follow these steps:

1. Go to the Junos Software Technical Publications index page for the release that you are using:

   http://www.juniper.net/techpubs/software/junos

2. Click the **Enterprise-Specific MIBs and Traps link** on the Junos Software Technical Publications index page.
3. Click the TAR or ZIP link under the TAR/ZIP column of the Enterprise MIBs row to download the Junos MIB package.
4. Extract the TAR or ZIP file by using an appropriate utility.
5. Load the standard MIB files (from the `StandardMibs` folder) in the following order:

   Standard MIBs: Some of the MIB compilers that are commonly used have the standard MIBs preloaded on them. If the standard MIBs are already loaded on the MIB compiler that you are using, skip this step and the next one, and proceed to step 7.

   ```
   a. mib-SNMPv2-SMI.txt
   b. mib-SNMPv2-TC.txt
   c. mib-IANAifType-MIB.txt
   d. mib-IANA-RTPROTO-MIB.txt
   e. mib-rfc1907.txt
   f. mib-rfc2011a.txt
   g. mib-rfc2012a.txt
   h. mib-rfc2013a.txt
   i. mib-rfc2863a.txt
   ```

6. Load the remaining standard MIB files.

   Attention: You must follow the order specified in this procedure. Also ensure that all standard MIBs are loaded before you load the enterprise-specific MIBs. Dependencies might exist that require a particular MIB to be present on the compiler before loading another MIB. Such dependencies are listed in the IMPORT section of the MIB file.

7. After loading the standard MIBs, load the Juniper Networks enterprise-specific SMI MIB, `mib-jnx-smi.txt`, and the following optional SMI MIBs based on your requirements:
   - Optional: The `mib-jnx-js-smi.txt` file for Juniper Security MIB tree objects
   - Optional: The `mib-jnx-ex-smi.txt` file for E Model Ethernet switches
   - Recommended: The `mib-jnx-exp.txt` file for the experimental MIB objects from Juniper Networks
8. Load the remaining enterprise-specific MIBs from the `JuniperMibs` folder.

Undefined object when loading an MIB file: While loading an MIB file, the compiler might return an error message indicating that an object is undefined. In this case, open the MIB file using a text editor and ensure that all the MIB files that are listed in the IMPORT section are loaded on the compiler. If any of the MIB files listed in the IMPORT section is not loaded on the compiler, load that MIB file, and then try to load the MIB file that failed to load.

For example, the enterprise-specific PING MIB, `mib-jnx-ping.txt`, has dependencies on RFC 2925, DiSMAN-PING-MIB, `mib-rfc2925a.txt`. If you try to load `mib-jnx-ping.txt` before loading `mib-rfc2925a.txt`, the compiler returns an error message indicating that certain objects in `mib-jnx-ping.txt` are undefined. Load `mib-rfc2925a.txt`, and then try to load `mib-jnx-ping.txt`. The enterprise-specific PING MIB, `mib-jnx-ping.txt`, then loads without any issue.

13.6 Using sFlow to monitor on a j-type e-series Ethernet switch

sFlow technology is a monitoring technology for high-speed switched or routed networks. sFlow monitoring technology randomly samples network packets and sends the samples to a monitoring station. You can configure sFlow technology on an IBM j-type e-series Ethernet switch to continuously monitor traffic at wire speed on all interfaces simultaneously.

sFlow technology uses the following two sampling mechanisms:

Packet-based sampling Samples one packet out of a specified number of packets from an interface enabled for sFlow technology.

Time-based sampling Samples interface statistics at a specified interval from an interface enabled for sFlow technology.

The sampling information is used to create a network traffic visibility picture. The Junos software fully supports the sFlow version 5 standard described at the sFlow.org website at:

http://www.sflow.org

Raw packet headers: sFlow technology on j-type e-series Ethernet switches samples only raw packet headers. A raw Ethernet packet is the complete Layer 2 network frame.

An sFlow monitoring system consists of an sFlow agent embedded in the switch and a centralized collector. The two main activities of the sFlow agent are random sampling and statistics gathering. It combines interface counters and flow samples and sends them across the network to the sFlow collector.

The IBM j-type e-series Ethernet switches adopt the distributed sFlow architecture. The sFlow agent has two separate sampling entities that are associated with each Packet Forwarding Engine. These sampling entities are known as *subagents*. Each subagent has a unique ID that is used by the collector to identify the data source. A subagent has its own independent state and forwards its own sample messages to the sFlow agent. The sFlow agent is responsible for packaging the samples into datagrams and sending them to the sFlow collector. Because sampling is distributed across subagents, the protocol overheads associated with sFlow are reduced at the collector. If the mastership assignment changes in a Virtual Chassis setup, the sFlow technology continues to function.

Junos software support of sFlow version 5: Junos software on IBM j-type e-series Ethernet switches supports sFlow version 5.

The sFlow collector uses the IP address of the sFlow agent to determine the source of the sFlow data. The IP address assigned to the agent is based on the following interfaces configured on the switch (in the order shown):

1. Virtual management Ethernet (VME) interface
2. Management Ethernet interface

If any of these interfaces has not been configured, the IP address of any Layer 3 interface or the routed VLAN interface (RVI) is used as the IP address for the agent. At least one interface must be configured for an IP address to be assigned to the agent.

sFlow datagram in EX Series switch implementation: In an EX Series switch implementation, the sFlow datagram cannot be routed over the management Ethernet interface (me0) or virtual management interface (vme0). It only can be exported over the network Gigabit Ethernet or 10 Gigabit Ethernet ports using valid route information in the routing table.

sFlow data can be used to provide network traffic visibility information. The IP address to be assigned to source data can be configured. If it has not been configured, the IP address of the configured Gigabit Ethernet interface, 10 Gigabit Ethernet interface, or the RVI is used as the source IP address. Infrequent sampling flows are not reported in the sFlow information, but over time, the majority of flows are reported. Based on a defined sampling rate, 1 out of N packets is captured and sent to the collector. This type of sampling does not provide a 100% accurate result in the analysis, but it does provide a result with quantifiable accuracy. A polling interval defines how often the sFlow data for a specific interface is sent to the collector, but an sFlow agent is free to schedule polling.

IBM j-type e-series Ethernet switches use adaptive sampling to ensure both sampling accuracy and efficiency. *Adaptive sampling* is a process of monitoring the overall incoming traffic rate on the network device. The process includes providing intelligent feedback to interfaces to dynamically adapt their sampling rate to the traffic conditions. Interfaces on which incoming traffic exceeds the system threshold are penalized so that all violations can be regulated without affecting the traffic on other interfaces.

Every 5 seconds, the agent checks interfaces to get the number of samples, and interfaces are grouped based on the slot to which they belong. The top five interfaces that produce the highest samples are selected. By using the binary backoff algorithm, the sampling load on these top five interfaces is reduced to half and adjusted on interfaces that have a lower sample rate. Therefore, when the processor limit is reached, the sampling rate is adapted such that it does not load the processor any further. If the switch is rebooted, the adaptive sample rate is reset to the user configured sample rate. Also, if you modify the sample rate, the adaptive sample rate changes.

The advantage of adaptive sampling is that the switch continues to operate at its optimum level even when the traffic patterns in the interfaces have a change. You do not need to make any changes. Because the sampling rate is adapted dynamically based on the network conditions, the resources are use optimally, resulting in a high performance network.

Graceful restart: Because sFlow technology on IBM j-type e-series Ethernet switches does not support graceful restart, when a graceful restart occurs, the adaptive sample rate is set to the user configured sample rate.

13.6.1 sFlow components

An sFlow monitoring system consists of an sFlow agent embedded in the switch and a centralized collector. The sFlow agent runs on the switch. It combines interface counters and flow samples and sends them across the network to the sFlow collector. Figure 13-2 shows the basic elements of the sFlow system.

Figure 13-2 sFlow components

13.6.2 Configuring sFlow technology

To configure sFlow technology, include the statements in the following commands at the `[edit protocols sflow]` hierarchy level.

Set the IP address of the collector and the port it will be listening on:

```
[edit protocols sflow]
root# set collector address udp-port udp-port
```

Maximum collectors: You can configure a maximum of four collectors.

Set the interfaces that you want to collect the sFlow data:

```
[edit protocols sflow]
root# set interfaces interface
```

Restrictions: You cannot enable sFlow technology on a Layer 3 VLAN-tagged interface. You cannot enable sFlow technology on a link aggregation group (LAG) interface, that is, an aggregated Ethernet interface with a name such as `ae0`. You can enable sFlow technology on the member interfaces that make up the LAG.

The *polling interval* is the interval between each port statistic polling update message, which can range from 0 to 3600 seconds:

```
[edit protocols sflow]
root# set polling-interval interval
```

Polling interval: The polling interval can be specified as a global parameter also. Specify 0 if you do not want to poll the interface.

The *sample rate* means that one out of N packets in the traffic stream will be sampled, which can be different for various interfaces. The range of sample rate is from 100 to 1 million.

```
[edit protocols sflow]
root#set sample-rate rate
```

13.6.3 Checking the results of a configuration

To confirm that the configuration is correct, use the **`show`** command at the `[edit protocols sflow]` hierarchy level:

```
[edit protocols sflow]
root# show
polling-interval 20;
sample-rate 1000;
collector 172.16.1.32 {
udp-port 6343;
}
interfaces ge-2/0/5.0;
```

The output shows that sFlow technology is enabled and specifies the values for the sample rate, sample limit, and polling interval.

13.6.4 Summary of the sFlow configuration

Example 13-9 shows the statements required to set sFlow monitoring on an IBM j-type e-series Ethernet switch.

Example 13-9 The sFlow configuration

```
Configure the IP address of the collector:
[edit protocols sflow]
root# set collector 172.16.1.32 <----Configure the IP address of the collector

[edit protocols sflow]
root# set collector udp-port 6343 <----Configure the UDP port of the collector.
                                        The default UDP port assigned is 6343.
[edit protocols sflow]
root# set interfaces ge-2/0/5.0 <-----Enable sFlow technology on a specific
                                        interface.
[edit protocols sflow]
root# set polling-interval 20 <-----Specify how often the sFlow agent polls the
                                     interface.
[edit protocols sflow]
root# set sample-rate 1000 <----Specify the rate at which packets must be sampled.

[edit protocols sflow]
ibm@J48E-VC# show
polling-interval 20;
sample-rate 1000;
collector 172.16.1.32 {
    udp-port 6343;
}
interfaces ge-2/0/5.0;
```

13.6.5 Viewing the collected data

With the network data traffic stream sending, the sFlow sampling data records and counter statistics record have been received on the remote collector as illustrated in Figure 13-3 on page 463. This figure shows incoming sFlow packets captured by using version 1.2.8 of the Wireshark tool, which is available from the following website:

http://www.wireshark.org

Figure 13-3 Incoming sFlow packets

The collected data can then be analyzed and displayed (Figure 13-4) in an application such as Scrutinizer NetFlow and sFlow Analyzer version 7.6.2. This software application is available at not cost and provides detailed network utilization information for the hosts and applications using the most bandwidth. You can find this software application at the following web page:

http://www.plixer.com/

Figure 13-4 SFlow Analyzer

13.7 Configuring system log messages

The Junos software generates system log messages, also called *syslog messages*, to record events that occur on the routing platform, such as the following events:

- Routine operations, such as the creation of an OSPF protocol adjacency or a user login into the configuration database
- Failure and error conditions, such as a failure to access a configuration file or unexpected closure of a connection to a peer process
- Emergency or critical conditions, such as routing platform power-down because of excessive temperature

Each system log message identifies the Junos software process that generated the message and briefly describes the operation or error that occurred. This reference provides more detailed information about each system log message. When applicable, it describes the possible causes of the message and action you can take to correct error conditions.

Configuration hierarchy: The configuration hierarchy in this chapter applies to Junos software processes and libraries, not to the services on a PIC, such as the Adaptive Services PIC. For information about configuring system logging for PIC services, see the *JUNOS Services Interfaces Configuration Guide*, GA32-0707.

13.7.1 Minimum and default system logging configuration

For information about the minimum and default system log settings on routing and switching platforms that run the Junos software, see the following sections:

- Minimum system logging configuration
- Default system log settings

Minimum system logging configuration

To record or view system log messages, you must include the **`syslog`** statement at the `[edit system]` hierarchy level. Specify at least one destination for the messages, as described in Table 13-4. For more information about the configuration statements, see “Configuring system logging for a single-chassis system” on page 466.

Table 13-4 Minimum configuration statements for system logging

Destination	Minimum configuration statements

File	`edit system syslog]`
	`file filename {`
	`facility severity ;`
	`}`

Terminal session of one, several, or all users	`[[edit system syslog]`
	`user (username | *) {`
	`facility severity ;`
	`}`

Routing platform console	`[edit system syslog]`
	`console {`
	`facility severity ;`
	`}`

Destination	Minimum configuration statements

Remote machine or the other Routing Engine on the routing platform	[edit system syslog]
	host (hostname | other-routing-engine) {
	facility severity ;
	}

Default system log settings

Table 13-5 summarizes the default system log settings that apply to all platforms that run the Junos software. It specifies which statement to include in the configuration to override the default value.

Table 13-5 Default system logging settings

Setting	Default	Overriding statement	Instructions
Alternative facility for message forwarded to a remote machine	For change-log: local6 For conflict-log: local5 For dfc: local1 For firewall: local3 For interactive-commands: local7 For pfe: local4	[edit system syslog] **host hostname {** **facility-override** *facility*; **}**	"Changing the alternative facility name for remote messages" on page 471
Format of messages logged to a file	Standard Junos format, based on the UNIX format	[edit system syslog] **file filename {** **structured-data;** **}**	"Logging messages in structured-data format" on page 469
Maximum number of files in the archived set	10	[edit system syslog] **archive {** **files** *number*; **}** **file** *filename* **{** **archive {** **files** *number* ; **}** **}**	"Configuring the size and number of log files" on page 474
Maximum size of log file	IBM j-type m-series 1 MB	[edit system syslog] **archive {** **size** *size*; **}** **file** *filename* **{** **archive {** **size** *size*; **}** **}**	"Configuring the size and number of log files" on page 474
Timestamp format	Month, date, hour, minute, second, for example: Aug 21 12:36:30	[edit system syslog] **time-format** *format*;	"Including the year or millisecond in time stamps" on page 477
Users who can read log files	root user and users with the Junos maintenance permission	[edit system syslog] **archive {** **world-readable;** **}** **file** *filename* **{** **archive {** **world-readable;** **}** **}**	"Configuring the size and number of log files" on page 474

In addition, a message is logged by default when a process running in the kernel consumes 500 or more consecutive milliseconds of CPU time. To view the message, you must configure at least one destination for messages as described in “Minimum system logging configuration” on page 464.

To log the message on an IBM j-type m-series or e-series platform, include the **kernel info** statement at the appropriate hierarchy level:

```
[edit system syslog]
root# set [console | file filename | host destination | user username] kernel
info}
```

Configuring system logging for a single-chassis system

The Junos system logging utility is similar to the UNIX **syslogd** utility. Each system log message belongs to a facility, which groups related messages. Each message is also pre-assigned a severity level, which indicates how seriously the triggering event affects routing platform functions. You always specify the facility and severity of the messages to include in the log. For more information, see “Specifying the facility and severity of messages to include in the log” on page 467.

You direct messages to one or more destinations by including the appropriate statement at the [edit system syslog] hierarchy level:

- To a named file in a local file system, by including the **file** statement. See “Directing messages to a log file” on page 468.
- To the terminal session of one or more specific users, or all users, when they are logged in to the routing platform, by including the **user** statement. See “Directing messages to a user terminal” on page 470.
- To the routing platform console, by including the **console** statement. See “Directing messages to the console” on page 470.
- To a remote machine that is running the **syslogd** utility or to the other Routing Engine on the routing platform, by including the **host** statement. See “Directing messages to a remote machine or the other Routing Engine” on page 470.

By default, messages are logged in a standard format, which is based on a UNIX system log format. For detailed information, see “Interpreting system log messages” on page 481. You can alter the content and format of logged messages in the following ways:

- In Junos version 8.3 and later, you can log messages to a file in a structured-data format instead of the standard Junos format. A structured-data format provides more information without adding significant length and makes it easier for automated applications to extract information from the message. For more information, see “Logging messages in structured-data format” on page 469.
- A message facility and severity level are together referred to as its priority. By default, the standard Junos format for messages does not include priority information. (Structured-data format includes a priority code by default.) To include priority information in standard-format messages directed to a file or a remote destination, include the **explicit-priority** statement. For more information, see “Including priority information in system log messages” on page 475.
- By default, the standard Junos format for messages specifies the month, date, hour, minute, and second when the message was logged. You can modify the time stamp on standard-format messages to include the year, the millisecond, or both. (The structured-data format specifies the year and millisecond by default.) For more information, see “Including the year or millisecond in time stamps” on page 477.

- When directing messages to a remote machine, you can specify the IP address that is reported in messages as their source. You can also configure features that make it easier to separate Junos-specific messages or messages generated on particular routing platforms. For more information, see “Directing messages to a remote machine or the other Routing Engine” on page 470.
- The predefined facilities group related messages, but you can also use regular expressions to specify more exactly which messages from a facility are logged to a file, a user terminal, or a remote destination. For more information, see “Using regular expressions to refine the set of logged messages” on page 478.

For a statement summary of the statements mentioned in this chapter, see the *JUNOS Software System Basics and Services Command Reference,* GA32-0671.

For detailed information about configuring system logging, see the following sections:

- “Specifying the facility and severity of messages to include in the log” on page 467
- “Directing messages to a log file” on page 468
- “Directing messages to a user terminal” on page 470
- “Directing messages to the console” on page 470
- “Directing messages to a remote machine or the other Routing Engine” on page 470
- “Configuring the size and number of log files” on page 474
- “Including priority information in system log messages” on page 475
- “Including the year or millisecond in time stamps” on page 477
- “Using regular expressions to refine the set of logged messages” on page 478
- “Disabling the logging of a facility” on page 479

Specifying the facility and severity of messages to include in the log

Each system log message belongs to a facility, which groups messages that either are generated by the same source (such as a software process) or concern a similar condition or activity (such as authentication attempts). Each message is also pre-assigned a severity level, which indicates how seriously the triggering event affects routing platform functions.

When you configure logging for a facility and destination, you specify a severity level for each facility. Messages from the facility that are rated at that level or higher are logged to the following destination:

```
[edit system syslog]
root# set [console | file filename | host destination | user username] facility
severity
```

For more information about the destinations, see the following sections:

- “Directing messages to a log file” on page 468
- “Directing messages to a user terminal” on page 470
- “Directing messages to the console” on page 470
- “Directing messages to a remote machine or the other Routing Engine” on page 470.

To log messages that belong to more than one facility for a particular destination, specify each facility and associated severity as a separate statement within the set of statements for the destination. Table 13-6 lists the facilities that you can specify in configuration statements at the `[edit system syslog]` hierarchy level.

Table 13-6 Junos system logging facilities

Facility	Type of event or error
any	All (messages from all facilities)
authorization	Authentication and authorization attempts

Facility	Type of event or error
change-log	Changes to the Junos configuration
conflict-log	Specified configuration is invalid on the routing platform type
daemon	Actions performed or errors encountered by system processes
dfc	Events related to dynamic flow capture
firewall	Packet filtering actions performed by a firewall filter
ftp	Actions performed or errors encountered by the FTP process
interactive-commands	Commands issued at the Junos CLI prompt or invoked by a client application such as a JUNOScript or NETCONF client
kernel	Actions performed or errors encountered by the Junos kernel
pfe	Actions performed or errors encountered by the Packet Forwarding Engine
user	Actions performed or errors encountered by user-space processes

Table 13-7 lists the severity levels that you can specify in configuration statements at the `[edit system syslog]` hierarchy level. The levels from *emergency* through *info* are in order from highest severity, with the greatest effect on functioning, to the lowest severity.

Unlike the other severity levels, the *none* level disables logging of a facility instead of indicating how seriously a triggering event affects routing functions. For more information, see "Disabling the logging of a facility" on page 479.

Table 13-7 System log message severity levels

Severity level	Description
any	Includes all severity levels
none	Disables logging of the associated facility to a destination
emergency	System panic or other condition that causes the routing platform to stop functioning
alert	Conditions that require immediate correction, such as a corrupted system database
critical	Critical conditions, such as errors with the hard disk drive
error	Error conditions that generally have less serious consequences than errors in the emergency, alert, and critical levels
warning	Conditions that warrant monitoring
notice	Conditions that are not errors but might warrant special handling
info	Events or nonerror conditions of interest

Directing messages to a log file

To direct system log messages to a file in the `/var/log` directory of the local Routing Engine, include the **`file`** statement at the `[edit system syslog]` hierarchy level:

```
[edit system syslog]
root# set file filename facility severity

[edit system syslog]
```

```
root# set file filename explicit-priority match "regular-expression"
structured-data brief
```

```
[edit system syslog]
root# set archive files number size size [world-readable | no-world-readable]
```

For the list of facilities and severity levels, see “Specifying the facility and severity of messages to include in the log” on page 467.

To prevent log files from growing too large, the Junos system logging utility by default writes messages to a sequence of files of a defined size. By including the **archive** statement, you can configure the number of files, their maximum size, and who can read them, for either all log files or a certain log file. For more information, see “Configuring the size and number of log files” on page 474.

For information about the following statements, see the following sections:

- For **explicit-priority**, see “Including priority information in system log messages” on page 475.
- For **match**, see “Using regular expressions to refine the set of logged messages” on page 478.
- For **structured-data**, see “Logging messages in structured-data format” on page 469.

Logging messages in structured-data format

In Junos 8.3 and later, you can log messages to a file in the structured-data format instead of the standard Junos format. The structured-data format provides more information without adding significant length and makes it easier for automated applications to extract information from a message.

The structured-data format complies with *The syslog Protocol*, *draft-ietf-syslog-protocol-23*, which is available at the following address:

http://tools.ietf.org/html/draft-ietf-syslog-protocol-23

This draft establishes a standard message format regardless of the source or transport protocol for logged messages.

To produce output messages to a file in the structured-data format, include the **structured-data** statement at the [edit system syslog file *filename*] hierarchy level:

```
[edit system syslog file filename]
root# facility severity structured-data brief
```

The optional **brief** statement suppresses the English-language text that is displayed by default at the end of a message to describe the error or event. For information about the fields in a structured-data format message, see “Displaying a log file from a single-chassis system” on page 480.

The structured format is used for all messages that are logged to the files that are generated by a Junos process or software library.

The explicit-priority and time-format statements: If you include the **explicit-priority** statement, the **time-format** statement, or both with the **structured-data** statement, they are ignored. These statements apply to the standard Junos system log format, not to the structured-data format.

Directing messages to a user terminal

To direct system log messages to the terminal session of one or more specific users, or all users, when they are logged in to the local Routing Engine, include the **user** statement at the [edit system syslog] hierarchy level:

```
[edit system syslog]
root# set user [username | *] facility severity match "regular-expression"
```

Specify one or more Junos user names, separating multiple values with spaces, or use the * to indicate all users who are logged in to the local Routing Engine.

For the list of facilities and severity levels, see “Specifying the facility and severity of messages to include in the log” on page 467. For information about the match statement, see “Using regular expressions to refine the set of logged messages” on page 478.

Directing messages to the console

To direct system log messages to the console of the local Routing Engine, include the **console** statement at the [edit system syslog] hierarchy level:

```
[edit system syslog]
root# set console facility severity
```

For the list of facilities and severity levels, see “Specifying the facility and severity of messages to include in the log” on page 467.

Directing messages to a remote machine or the other Routing Engine

To direct system log messages to a remote machine or to the other Routing Engine on the routing platform, include the **host** statement at the [edit system syslog] hierarchy level:

```
[edit system syslog]
root# host [hostname | other-routing-engine] facility severity explicit-priority
facility-override facility log-prefix string match "regular-expression";

[edit system syslog]
root# set source-address source-address
```

To direct system log messages to a remote machine, include the **host** *hostname* statement to specify the IP version 4 address, IP version 6 address, or fully qualified host name of the remote machine. The remote machine must be running the standard **syslogd** utility. Do not direct messages to another Juniper Networks routing platform. In each system log message directed to the remote machine, the host name of the local Routing Engine is displayed after the time stamp to indicate that it is the source for the message.

To direct system log messages to the other Routing Engine on a routing platform with two Routing Engines installed and operational, include the **host other-routing-engine** statement. The statement is not automatically reciprocal. Therefore, you must include it in the configuration of each Routing Engine if you want them to direct messages to each other. In each message directed to the other Routing Engine, the string *re0* or *re1* is displayed after the time stamp to indicate the source for the message.

For the list of facilities and severity levels to configure under the host statement, see “Specifying the facility and severity of messages to include in the log” on page 467.

To record facility and severity level information in each message, include the **explicit-priority** statement. For more information, see “Including priority information in system log messages” on page 475.

For information about the **match** statement, see “Using regular expressions to refine the set of logged messages” on page 478.

When directing messages to remote machines, you can include the **source-address** statement to specify the IP address of the routing platform that is reported in the messages as their source. In each host statement, you can also include the **facility-override** statement to assign an alternative facility and the **log-prefix** statement to add a string to each message. For more information, see the following sections:

- “Specifying an alternative source address for system log messages” on page 471
- “Changing the alternative facility name for remote messages” on page 471
- “Assigning an alternative facility” on page 473
- “Adding a text string to system log messages” on page 473
- “Adding a string” on page 474

Specifying an alternative source address for system log messages

To specify the routing platform that is reported in system log messages as their source when they are directed to a remote machine, include the **source-address** statement at the **[edit system syslog]** hierarchy level:

```
[edit system syslog]
root# set source-address source-address
```

where *source-address* is a valid IPv4 or IPv6 address configured on one of the routing platform interfaces. The address is reported in the messages that are directed to all remote machines specified in **host** *hostname* statements at the [edit system syslog] hierarchy level. It is not reported in the messages that are directed to the other Routing Engine.

Changing the alternative facility name for remote messages

Some facilities assigned to messages logged on the local routing platform have Junos-specific names (see Table 13-6 on page 467). In the configuration, a remote machine designated at the [edit system syslog host *hostname*] hierarchy level is not a Juniper Networks routing platform. Therefore, its **syslogd** utility cannot interpret the Junos-specific names. To enable the standard **syslogd** utility to handle messages from these facilities, when messages are directed to a remote machine a standard *localX* facility name is used instead of the Junos-specific facility name.

Table 13-8 lists the default alternative facility name that is used for each Junos-specific facility name. For facilities that are not listed, the default alternative name is the same as the local facility name.

Table 13-8 Default facilities for messages directed to a remote destination

Junos-specific local facility	Default facility when directed to remote destination
change-log	local6
conflict-log	local5
dfc	local1
firewall	local3
interactive-commands	local7
pfe	local4

The **syslogd** utility on a remote machine handles all messages for a facility in the same way, regardless of the source of the message, the Juniper Networks routing platform, or the

remote machine itself. For example, the following statements in the configuration of the routing platform called `local-router` direct messages from the authorization facility to the remote machine called `monitor.mycompany.com`:

```
[edit system syslog]
root# set host monitor.mycompany.com authorization info
```

The default alternative facility for the local **authorization** facility is also **authorization**. The **syslogd** utility on monitor might be configured to write messages that belong to the **authorization** facility to the `/var/log/auth-attempts` file. In this case, the file contains both messages that are generated when users log in to **local-router** and the messages generated when users log in to **monitor**. Although the name of the source machine is displayed in each system log message, the mixing of messages from multiple machines can make it more difficult to analyze the contents of the `auth-attempts` file.

To make it easier to separate the messages from each source, you can assign an alternative facility to all messages that are generated on **local-router** when they are directed to **monitor**. You can then configure the **syslogd** utility on monitor to write messages with the alternative facility to a different file from the messages that are generated on **monitor** itself.

To change the facility used for all messages directed to a remote machine, include the **facility-override** statement at the [`edit system syslog host` *hostname*] hierarchy level:

```
[edit system syslog host hostname]
root# facility severity facility-override facility
```

In general, specify an alternative facility that is not already in use on the remote machine, such as one of the *localX* facilities. On the remote machine, you must also configure the **syslogd** utility to handle the messages in the desired manner.

Table 13-9 lists the facilities that you can specify in the **facility-override** statement.

Table 13-9 Facilities for the facility-override statement

Facility	Description
authorization	Authentication and authorization attempts
daemon	Actions performed or errors encountered by system processes
ftp	Actions performed or errors encountered by the FTP process
kernel	Actions performed or errors encountered by the Junos kernel
local0	Local facility number 0
local1	Local facility number 1
local2	Local facility number 2
local3	Local facility number 3
local4	Local facility number 4
local5	Local facility number 5
local6	Local facility number 6
local7	Local facility number 7
user	Actions performed or errors encountered by user-space processes

Do not include the **facility-override** statement at the [edit system syslog host other-routing-engine] hierarchy level. It is not necessary to use alternative facility names when directing messages to the other Routing Engine, because its Junos system logging utility can interpret the Junos-specific names.

Assigning an alternative facility

To log all messages that are generated on the local routing platform at the error level or higher to the **local0** facility on the remote machine called **monitor.mycompany.com**, use the following command:

```
[edit system syslog]
root# set host monitor.mycompany.com any error

[edit system syslog]
root# set host monitor.mycompany.com facility-override local0
```

You can configure routing platforms in California and New York to send messages to a single remote machine called **central-logger.mycompany.com**. The messages from California are assigned to alternative facility **local0**. The messages from New York are assigned to alternative facility **local2**. The following examples illustrate to configure the routing platforms:

- Configure California routing platforms to aggregate messages in the **local0** facility:

  ```
  [edit system syslog]
  root@cal# set host central-logger.mycompany.com change-log info

  [edit system syslog]
  root@cal# set host central-logger.mycompany.com facility-override local0
  ```

- Configure New York routing platforms to aggregate messages in the **local2** facility:

  ```
  [edit system syslog]
  root@ny# set host central-logger.mycompany.com change-log info

  [edit system syslog]
  root@ny# set host central-logger.mycompany.com facility-override local2
  ```

On **central-logger**, you can then configure the system logging utility to write messages from the **local0** facility to the `california-config` file and the messages from the **local2** facility to the `new-york-config` file.

Adding a text string to system log messages

To add a text string to every system log message directed to a remote machine or to the other Routing Engine, include the **log-prefix** statement at the [edit system syslog host] hierarchy level:

```
[edit system syslog host [hostname | other-routing-engine]]
root# set facility severity

[edit system syslog host [hostname | other-routing-engine]]
root# set log-prefix string
```

Alphanumeric and special characters: The string can contain any alphanumeric or special character except the equal sign (=) and the colon (:). It also cannot include the space character. Do not enclose the string in quotation marks (" ") in an attempt to include spaces in it.

The Junos system logging utility automatically appends a colon and a space to the specified string. The string is inserted after the identifier for the Routing Engine that generated the message.

Adding a string

Add the **J06M** string to all messages to indicate that the router is an IBM j-type m-series Ethernet router, and direct the messages to the **hardware-logger.mycompany.com** remote machine:

```
[edit system syslog]
root# set host hardware-logger.mycompany.com any info

[edit system syslog]
root# set host hardware-logger.mycompany.comlog-prefix M120}
```

When these configuration statements are included on an IBM j-type m-series Ethernet router called **Per_Router**, a message similar to the following example is displayed in the system log on **hardware-logger.mycompany.com**:

```
May17 09:33:23 Per_router J06M: mgd[477]: UI_CMDLINE_READ_LINE: user 'root',
command 'run show version'
```

Configuring the size and number of log files

To prevent log files from growing too large, the Junos system logging utility, by default, writes messages to a sequence of files of a defined size. The files in the sequence are referred to as *archive files* to distinguish them from the active file to which messages are currently being written.

When an active log file, called `logfile`, reaches the maximum size, the logging utility closes the file, compresses it, and names the compressed archive file `logfile.0.gz`. The logging utility then opens and writes to a new active file called `logfile`. When the new logfile reaches the configured maximum size, `logfile.0.gz` is renamed `logfile.1.gz`, and the new `logfile` is closed, compressed, and renamed `logfile.0.gz`.

By default, the logging utility creates up to 10 archive files in this manner. When the maximum number of archive files is reached, each time the active file reaches the maximum size, the contents of the oldest archive file are lost and overwritten by the next oldest file. The logging utility, by default, also limits the users who can read log files to the root user and users who have the Junos maintenance permission.

You can include the **archive** statement to change the maximum size of each file, the number of archive files that are created, and who can read log files. To configure values that apply to all log files, include the **archive** statement at the `[edit system syslog]` hierarchy level:

```
[edit system syslog]
root# set archive files number size size [world-readable | no-world-readable]
```

To configure values that apply to a particular log file, include the **archive** statement at the `[edit system syslog file filename]` hierarchy level:

```
[edit system syslog file filename]
root# archive files number size size [world-readable | no-world-readable]
```

where:

files *number*	specifies the number of files to create before the oldest file is overwritten. The value can be from 1 through 1000.

size *size*	specifies the maximum size of each file. The value can be from 64 KB (64k) through 1 GB (1g). To represent MB, use the letter m after the integer. No space exists between the digits and the k, m, or g units letter.
world-readable	enables all users to read log files. To restore the default permissions, include the **no-world-readable** statement.

Including priority information in system log messages

The facility and severity level of a message are referred to as its *priority*. By default, messages that are logged in the standard Junos format do not include information about the priority. To include priority information in standard-format messages directed to a file, include the **explicit-priority** statement at the [edit system syslog file *filename*] hierarchy level:

```
[edit system syslog file filename]
root# set facility severity

[edit system syslog file filename]
root# set explicit-priority
```

> **Messages in the structure-data format:** Messages logged in the structured-data format include priority information by default. (The structured-data format is available in Junos Release 8.3 and later and for file destinations only.) If you include the **structured-data** statement at the [edit system syslog file *filename*] hierarchy level with the **explicit-priority** statement, the **explicit-priority** statement is ignored, and messages are logged in the structured-data format.
>
> For information about the **structured-data** statement, see "Logging messages in structured-data format" on page 469. For information about the contents of a structured-data message, see "Displaying a log file from a single-chassis system" on page 480.

To include priority information in messages directed to a remote machine or the other Routing Engine, include the **explicit-priority** statement at the [edit system syslog host [*hostname* | *other-routing-engine*]] hierarchy level:

```
[edit system syslog host [hostname | other-routing-engine]]
root# set facility severity

[edit system syslog host [hostname | other-routing-engine]]
root# set explicit-priority
```

The priority recorded in a message always indicates the original, local facility name. If the **facility-override** statement is included for messages directed to a remote destination, the Junos system logging utility still uses the alternative facility for the messages themselves when directing them to the remote destination. For more information about alternative facilities, see "Changing the alternative facility name for remote messages" on page 471.

When the **explicit-priority** statement is included, the Junos logging utility adds a prefix to codes for the facility name and severity level to the message tag name, if the message has one:

facility-severity[-TAG]

The tag is a unique identifier assigned to some Junos system log messages. For more information, see "Displaying a log file from a single-chassis system" on page 480, and "Displaying and interpreting system log messages" on page 480.

Table 13-10 lists the facility codes that can be displayed in system log messages and maps them to facility names.

Junos facility name: If the second column in Table 13-10 does not include the Junos facility name for a code, the facility cannot be included in a statement at the `[edit system syslog]` hierarchy level. The Junos software might use the facilities in Table 13-10, and others that are not listed, when reporting on internal operations

Table 13-10 Facility codes reported in priority information

Code	Junos facility name	Type of event or error
AUTH	authorization	Authentication and authorization attempts
AUTHPRIV		Authentication and authorization attempts that can be viewed by superusers only
CHANGE	change-log	Changes to the Junos configuration
CONFLICT	conflict-log	Specified configuration is invalid on the routing platform type
CONSOLE		Messages written to `/dev/console` by the kernel console output driver
CRON		Actions performed or errors encountered by the cron process
DAEMON	daemon	Actions performed or errors encountered by system processes
DFC	dfc	Actions performed or errors encountered by the dynamic flow capture process
FIREWALL	firewall	Packet filtering actions performed by a firewall filter
FTP	ftp	Actions performed or errors encountered by the FTP process
INTERACT	interactive-commands	Commands issued at the Junos CLI prompt or invoked by a client application such as a JUNOScript or NETCONF client
KERN	kernel	Actions performed or errors encountered by the Junos kernel
NTP		Actions performed or errors encountered by the Network Time Protocol (NTP) process
PFE	pfe	Actions performed or errors encountered by the Packet Forwarding Engine
SYSLOG		Actions performed or errors encountered by the Junos system logging utility
USER	user	Actions performed or errors encountered by user-space processes

Table 13-11 lists the numeric severity codes that can be displayed in system log messages and maps them to severity levels.

Table 13-11 Numeric codes for severity levels reported in priority information

Numeric code	Severity level	Description
0	emergency	System panic or other condition that causes the routing platform to stop functioning
1	alert	Conditions that require immediate correction, such as a corrupted system database
2	critical	Critical conditions, such as errors with the hard disk drive
3	error	Error conditions that generally have less serious consequences than errors in the emergency, alert, and critical levels
4	warning	Conditions that warrant monitoring
5	notice	Conditions that are not errors but might warrant special handling
6	info	Events or nonerror conditions of interest
7	debug	Software debugging messages **Displayed messages:** These messages are displayed only if a technical support representative has instructed you to configure this severity level.

In Example 13-10, the `CHASSISD_PARSE_COMPLETE` message belongs to the daemon facility and is assigned a severity level 6 (info).

Example 13-10 Log message with explicit-priority

```
Jul 25 15:36:30 router1 chassisd[522]:
%DAEMON-6-CHASSISD_PARSE_COMPLETE: Using new configuration
```

When the **`explicit-priority`** statement is not included, the priority is not displayed in the message (Example 13-11).

Example 13-11 Log message with no explicit-priority

```
Jul 25 15:36:30 router1 chassisd[522]: CHASSISD_PARSE_COMPLETE: Using new
configuration
```

For more information about message formatting, see “Displaying and interpreting system log messages” on page 480.

Including the year or millisecond in time stamps

By default, the time stamp recorded in a standard-format system log message specifies the month, date, hour, minute, and second when the message was logged, as in the following example:

```
May 17 09:36:30
```

To include the year, the millisecond, or both in the time stamp, include the **`time-format`** statement at the `[edit system syslog]` hierarchy level:

```
edit system syslog]
root# set time-format [year | millisecond | year millisecond]
```

The modified time stamp is used in messages that are directed to each destination that is configured by a **file**, **console**, or **user** statement at the [edit system syslog] hierarchy level, but not to destinations configured by a **host** statement.

The following example illustrates the format for a time stamp that includes both the millisecond (401) and the year (1992):

```
May 17 09:36:30.401 1992
```

Messages in the structured-data format: Messages that are logged in structured-data format, which is available in Junos Release 8.3 and later for file destinations, include the year and millisecond by default. If you include the **structured-data** statement at the [edit system syslog file *filename*] hierarchy level with the **time-format** statement, the **time-format** statement is ignored, and messages are logged in the structured-data format.

For information about the **structured-data** statement, see "Logging messages in structured-data format" on page 469. For information about the information in a structured-data message, see "Displaying a log file from a single-chassis system" on page 480

Using regular expressions to refine the set of logged messages

The predefined facilities group related messages. However, you can also use regular expression matching to specify more exactly which messages from a facility are logged to a file, a user terminal, or a remote destination.

To specify the text string that must, or must not, be displayed in a message for the message to be logged to a destination, include the match statement, and specify the regular expression which the text string must match:

match *"regular-expression"*

You can include this statement at the following hierarchy levels:

- [edit system syslog file filename] for a file
- [edit system syslog user [*username* | *]] for the terminal session of one or all users
- [edit system syslog host [hostname | *other-routing-engine*]] for a remote destination

In specifying the regular expression, use the notation that is defined in POSIX Standard 1003.2 for extended (modern) UNIX regular expressions. Explaining regular expression syntax is beyond the scope of this document, but POSIX standards are available from the Institute of Electrical and Electronics Engineers (IEEE) at the following website:

http://www.ieee.org

Table 13-12 on page 479 specifies which character or characters are matched by some of the regular expression operators that you can use in the match statement. In the descriptions, the term *term* refers to a single alphanumeric character or a set of characters enclosed in square brackets, parentheses, or braces.

Match statement: The match statement is not case-sensitive.

Table 13-12 Regular expression operators for the match statement

Operator	Matches
. (period)	One instance of any character except the space.
* (asterisk)	Zero or more instances of the immediately preceding term.
+ (plus sign)	One or more instances of the immediately preceding term.
? (question mark)	Zero or one instance of the immediately preceding term.
| (pipe)	One of the terms that is displayed on either side of the pipe operator.
! (exclamation mark)	Any string except the one specified by the expression, when the exclamation point is displayed at the start of the expression. Use of the exclamation point is Junos software-specific.
^ (caret)	Start of a line when the caret is displayed outside square brackets. One instance of any character that does not follow it within square brackets is when the caret is the first character inside square brackets.
$ (dollar sign)	End of a line.
[] (paired square brackets)	One instance of one of the enclosed alphanumeric characters. To indicate a range of characters, use a hyphen (-) to separate the beginning and ending characters of the range. For example, [a-z0-9] matches any letter or number.
() (paired parentheses)	One instance of the evaluated value of the enclosed term. Parentheses are used to indicate the order of evaluation in the regular expression.

Filter messages that belong to the **interactive-commands** facility, directing those expressions that include the string **configure** to the terminal of the root user:

```
[edit system syslog]
root# set user root interactive-commands any

[edit system syslog]
root# set user root match ".*configure.*"
```

Messages, such as the following example, are displayed on the root user's terminal when a user issues a configure command to enter configuration mode:

```
timestamp router-name mgd[PID]: UI_CMDLINE_READ_LINE: User 'user', command
'configure private'
```

Disabling the logging of a facility

To disable the logging of messages that belong to a particular facility, include the *facility* **none** statement in the configuration. This statement is useful, for example, when you want to log messages that have the same severity level and belong to all but a few facilities. Instead of including a statement for each facility you want to log, you can include the **any severity** statement and then a *facility* **none** statement for each facility that you do not want to log.

Example 13-12 shows the logging of all messages at the error level or higher to the console, except for messages from the daemon and kernel facilities. Messages from those facilities are logged to the `/var/log/internals` file instead.

Example 13-12 Disabling the logging of a facility

```
[edit system syslog]
root# set console any error

[edit system syslog]
root# set console daemon none

[edit system syslog]
root# set console kernel none

[edit system syslog]
root# set file internals daemon info

[edit system syslog]
root# set file internals kernel info
```

13.7.2 Displaying and interpreting system log messages

You can display a log file and interpret the contents of system log messages as explained in the following sections:

- Displaying a log file from a single-chassis system
- Interpreting system log messages
- Format of the message-source field
- Viewing a log file
- Getting help for system log messages
- Viewing system log message descriptions

For more information about the commands mentioned in this section, see the *JUNOS System Basics and Services Command Reference*, GA32-0671-02.

Displaying a log file from a single-chassis system

To see a log file stored on a single-chassis system, enter the Junos CLI operational mode and issue either of the following commands:

```
user@host> show log log-filename
user@host> file show log-file-pathname
```

By default, the commands show the file stored on the local Routing Engine. To view the file stored on a particular Routing Engine, prefix *file-* or *pathname* with the string **re0** or **re1** and a colon. The following examples display the `/var/log/messages` file that is stored on the Routing Engine in slot 1:

```
user@host> show log re1:messages
user@host> file show re1:/var/log/messages
```

For information about the fields in a log message, see 13.7.2, “Displaying and interpreting system log messages” on page 480.

Interpreting system log messages

The fields in a message written to the system log depend on whether the message was generated by a Junos process or subroutine library, or by services on a PIC. They also depend on whether it is in standard format or structured-data format. For more information, see the following sections:

- Interpreting messages generated in the structured-data format
- Interpreting messages generated in the standard format by a Junos process or library
- Interpreting messages generated in the standard format by services on a PIC

Interpreting messages generated in the structured-data format

Beginning in Junos Release 8.3, when the **`structured-data`** statement is included in the configuration for a log file, Junos processes and software libraries write messages to the file in the structured-data format instead of the standard Junos format. For information about the **`structured-data`** statement, see “Logging messages in structured-data format” on page 469.

The structured format makes it easier for automated applications to extract information from the message. In particular, the standardized format for reporting the value of variables, which are elements in the English-language message that vary depending on the circumstances that triggered the message, makes it easy for an application to extract those values. In the standard format, the variables are interspersed in the message text and not identified as variables.

The structured-data format for a message includes the following fields (which are displayed here on two lines for legibility reasons):

```
<priority code>version timestamp hostname process processID TAG
[junos@2636.platform variable-value-pairs] message-text
```

Table 13-13 describes the fields. If the system logging utility cannot determine the value in a particular field, a hyphen is displayed instead.

Table 13-13 Fields in structured-data messages

Field	Description	Examples
<priority code>	A number that indicates the facility and severity of the message. It is calculated by multiplying the facility number by 8 and then adding the numeric value of the severity. For a mapping of the numeric codes to facility and severity, see Table 13-14 on page 482.	<165> for a message from the PFE facility (facility=20) with severity notice (severity=5).
version	Version of the Internet Engineering Task Force (IETF) system logging protocol specification.	1 for the initial version
timestamp	The time when the message was generated, in one of two representations: ► *YYYY-MM-DDTHH:MM:SS.MSZ* is the year, month, day, hour, minute, second and millisecond in Universal Coordinated Time (UTC) ► *YYYY-MM-DDTHH:MM:SS.MS+/-HH:MM* is the year, month, day, hour, minute, second and millisecond in local time; the hour and minute that follows the + sign or - sign is the offset of the local time zone from UTC	2007-02-15T09:17:15.719Z is 9:17 AM UTC on 15 February 2007. 2007-02-15T01:17:15.719 -08:00 is the same time stamp expressed as Pacific Standard Time in the United States.
hostname	Name of the host that originally generated the message.	router1
process	Name of the Junos process that generated the message.	mgd

Field	Description	Examples
processID	UNIX process ID (PID) of the Junos process that generated the message.	3046
TAG	Junos system log message tag, which uniquely identifies the message. For a list of the tags described in this section, see *JUNOS Software System Log Messages Reference*, GA32-0675.	UI_DBASE_LOGOUT_EVENT
junos@2636.platform	An identifier for the type of hardware platform that generated the message. The junos@2636 prefix indicates that the platform runs the Junos software. It is followed by a dot-separated numeric identifier for the platform type. For a list of the identifiers, see *JUNOS Software System Log Messages Reference*, GA32-0675.	junos@2636.1.1.1.2.18 for the M120 router
variable-value-pairs	A variable-value pair for each element in the message-text string that varies depending on the circumstances that triggered the message. Each pair is displayed in the format **`variable = "value"`**.	username="regress"
message-text	The English-language description of the event or error, omitted if the **`brief`** statement is included at the `[edit system syslog file filename structured-data]` hierarchy level. For the text of each message, see *JUNOS Software System Log Messages Reference,* GA32-0675.	User 'regress' exiting configuration mode

By default, the structured-data version of a message includes English text at the end, as in the following example (which is displayed on multiple lines for legibility reasons):

```
<165>1 2007-02-15T09:17:15.719Z router1 mgd 3046 UI_DBASE_LOGOUT_EVENT
[junos@2636.1.1.1.2.18 username="regress"] User 'regress' exiting configuration
mode
```

When the **`brief`** statement is included at the `[edit system syslog file filename structured-data ]` hierarchy level, the English text is omitted, as in the following example:

```
<165>1 2007-02-15T09:17:15.719Z router1 mgd 3046 UI_DBASE_LOGOUT_EVENT
[junos@2636.1.1.1.2.18 username="regress"]
```

Table 13-14 maps the codes that are displayed in the priority-code field to facility and severity level.

Additional facility and severity codes: Not all of the facilities and severities listed in Table 13-14 can be included in statements at the `[edit system syslog]` hierarchy level. Some are used by internal processes. For a list of the facilities and severity levels that can be included in the configuration, see “Specifying the facility and severity of messages to include in the log” on page 467.

Table 13-14 Facility and severity codes in the priority-code field

Facility (number)	Severity emergency	alert	critical	error	warning	notice	info	debug
kernel (0)	1	1	2	3	4	5	6	7
user (1)	8	9	10	11	12	13	14	15
mail (2)	16	17	18	19	20	21	22	23

Facility (number)	Severity emergency	alert	critical	error	warning	notice	info	debug
daemon (3)	24	25	26	27	28	29	30	31
authorization (4)	32	33	34	35	36	37	38	39
syslog (5)	40	41	42	43	44	45	46	47
printer (6)	48	49	50	51	52	53	54	55
news (7)	56	57	58	59	60	61	62	63
uucp (8)	64	65	66	67	68	69	70	71
clock (9)	72	73	74	75	76	77	78	79
authorization-private (10)	80	81	82	83	84	85	86	87
ftp (11)	88	89	90	91	92	93	94	95
ntp (12)	96	97	98	99	100	101	102	103
security (13)	104	105	106	107	108	109	110	111
console (14)	112	113	114	115	116	117	118	119
local0 (16)	128	129	130	131	132	133	134	135
dfc (17)	136	137	138	139	140	141	142	143
local2 (18)	144	145	146	147	148	149	150	151
firewall (19)	152	153	154	155	156	157	158	159
pfe (20)	160	161	162	163	164	165	166	167
conflict-log (21)	168	169	170	171	172	173	174	175
change-log (22)	176	177	178	179	180	181	182	183
interactive-commands (23)	184	185	186	187	188	189	190	191

Interpreting messages generated in the standard format by a Junos process or library

The syntax of a standard-format message that is generated by a Junos software process or subroutine library depends on whether it includes priority information:

- When the **`explicit-priority`** statement is included at the [`edit system syslog file` *`filename`*] or [`edit system syslog host` [*`hostname`* | *`other-routing-engine`*]] hierarchy level, a system log message has the following syntax:

  ```
  timestamp message-source: %facility.severity.TAG: message-text
  ```

- When directed to the console or to users, or when the **`explicit-priority`** statement is not included for files or remote hosts, a system log message has the following syntax:

  ```
  timestamp message-source: TAG: message-text
  ```

Table 13-15 describes the message fields.

Table 13-15 Fields in standard-format messages generated by a Junos process

Field	Description
timestamp	Time at which the message was logged.
message-source	Identifier of the process or component that generated the message and the routing platform on which the message was logged. This field includes two or more subfields, depending on how system logging is configured. For more information, see “Format of the message-source field” on page 485.
facility	Code that specifies the facility to which the system log message belongs. For a mapping of codes to facility names, see Table 13-10 on page 476.
severity	Numeric code that represents the severity level that is assigned to the system log message. For a mapping of codes to severity names, see Table 13-11 on page 477.
TAG	Text string that uniquely identifies the message, in all uppercase letters and using the underscore (_) to separate words. The tag name begins with a prefix that indicates the generating software process or library. The entries in this reference are in alphabetical order by this prefix. Not all processes on a routing platform use tags. Therefore, this field is not always displayed. For a list of prefixes for the tags described in this reference, see *JUNOS Software System Log Messages Reference*, GA32-0675.
message-text	Text of the message. For the text for each message, see *JUNOS Software System Log Messages Reference*, GA32-0675.

Interpreting messages generated in the standard format by services on a PIC

Standard-format system log messages generated by services on a PIC, such as the Adaptive Services (AS) PIC, have the following syntax:

```
timestamp (FPC Slot fpc-slot, PIC Slot pic-slot) {service-set} [SERVICE]:
optional-string TAG: message-text
```

System logging services: System logging for services on PICs is not configured at the `[edit system syslog]` hierarchy level as discussed in this chapter. For configuration information, see the *JUNOS Software Services Interfaces Configuration Guide*, GA32-0707-02.

The (FPC Slot fpc-slot, PIC Slot pic-slot) field is displayed only when the standard system logging utility that runs on the Routing Engine writes the messages to the system log. When the PIC writes the message directly, the field is not displayed.

Table 13-16 describes the message fields.

Table 13-16 Fields in Messages Generated by a PIC

Field	Description
timestamp	Time at which the message was logged.
fpc-slot	Slot number of the Flexible PIC Concentrator (FPC) that houses the PIC that generated the message.
pic-slot	Number of the PIC slot on the FPC in which the PIC that generated the message resides.
service-set	Name of the service set that generated the message.

Field	Description
SERVICE	Code representing the service that generated the message. The codes include the following examples: ► FWNAT. Network Address Translation (NAT) service. ► IDS. Intrusion detection service.
optional-string	A text string that is displayed if the configuration for the PIC includes the log-prefix statement at the `[edit interfaces interface-name services-options syslog]` hierarchy level. For more information, see the *JUNOS Software Services Interfaces Configuration Guide,* GA32-0707-02.
TAG	Text string that uniquely identifies the message, in all uppercase letters and using the underscore (_) to separate words. The tag name begins with a prefix that indicates the generating PIC. The entries in this reference are in alphabetical order by this prefix.
message-text	Text of the message. For the text of each message, see *JUNOS Software System Log Messages Reference*, GA32-0675.

Format of the message-source field

The `message-source` field, as explained in "Interpreting messages generated in the standard format by a Junos process or library" on page 483, has two or more subfields. They depend on whether the message is logged on a single-chassis system or on a platform in a routing matrix. They also depend on whether message forwarding is configured in the routing matrix.

The message-source field on a single-chassis system

The format of the `message-source` field in a message on a single-chassis system depends on whether the message was generated on the local Routing Engine or the other Routing Engine, on a system with two Routing Engines that are installed and operational. Messages from the other Routing Engine are displayed only if its configuration includes the **`other-routing-engine`** statement at the `[edit system syslog host]` hierarchy level.

- When the local Routing Engine generated the message, there are two subfields:

  `hostname process[process-ID]`

- When the other Routing Engine generated the message, there are three subfields:

  `hostname reX process[process-ID]`

where:

`hostname`	is the host name of the local Routing Engine.
`process[process-ID]`	is the name and PID of the process that generated the message. If the *`reX`* field is also displayed, the process is running on the other Routing Engine. If a process does not report its PID, the [process-ID] part is not displayed.
`reX`	indicates that the other Routing Engine generated the message (*X* is 0 or 1).

Viewing a log file

You can view the contents of the `/var/log/messages` file stored on the local Routing Engine as shown in the following example. The `/var/log` directory is the default location for log files. Therefore, you do not need to include it in the file name. The `messages` file is a commonly configured destination for system log messages.

```
user@host> show log messages Apr 11 10:27:25 router1 mgd[3606]:
UI_DBASE_LOGIN_EVENT: User 'barbara' entering configuration mode
Apr 11 10:32:22 router1 mgd[3606]: UI_DBASE_LOGOUT_EVENT: User 'barbara' exiting
configuration mode
Apr 11 11:36:15 router1 mgd[3606]: UI_COMMIT: User 'root' performed commit: no
comment
Apr 11 11:46:37 router1 mib2d[2905]: SNMP_TRAP_LINK_DOWN: ifIndex 82,
ifAdminStatus up(1), ifOperStatus down(2), ifName at-1/0/0
```

View the contents of the `/var/log/processes` file, which was previously configured to include messages from the daemon facility, as shown in the following example. When issuing the **file show** command, you must specify the full path name of the file.

```
user@host> file show /var/log/processes Feb 22 08:58:24 router1 snmpd[359]:
SNMPD_TRAP_WARM_START: trap_generate_warm: SNMP trap: warm start
Feb 22 20:35:07 router1 snmpd[359]: SNMPD_THROTTLE_QUEUE_DRAINED:
trap_throttle_timer_handler: cleared all throttled traps
Feb 23 07:34:56 router1 snmpd[359]: SNMPD_TRAP_WARM_START: trap_generate_warm:
SNMP trap: warm start
Feb 23 07:38:19 router1 snmpd[359]: SNMPD_TRAP_COLD_START: trap_generate_cold:
SNMP trap: cold start
```

Display the contents of the `/var/log/processes` file when the **explicit-priority** statement is included at the `[edit system syslog file processes]` hierarchy level as shown in the following example:

```
user@host> file show /var/log/processes Feb 22 08:58:24 router1 snmpd[359]:
%DAEMON-3-SNMPD_TRAP_WARM_START: trap_generate_warm: SNMP trap: warm start
Feb 22 20:35:07 router1 snmpd[359]: %DAEMON-6-SNMPD_THROTTLE_QUEUE_DRAINED:
trap_throttle_timer_handler: cleared all throttled traps
Feb 23 07:34:56 router1 snmpd[359]: %DAEMON-3-SNMPD_TRAP_WARM_START:
trap_generate_warm: SNMP trap: warm start
Feb 23 07:38:19 router1 snmpd[359]: %DAEMON-2-SNMPD_TRAP_COLD_START:
trap_generate_cold: SNMP trap: cold start
```

Getting help for system log messages

Tag names in the system log message begin with a prefix that indicates which Junos software process, subroutine library, or service on a PIC generated the message. This section explains where you can find more help in understanding the messages with each prefix, both in this section and while you are using the CLI. For a list of the prefixes for messages described in this section, see *JUNOS Software System Log Messages Reference,* GA32-0675.

For information about displaying and interpreting messages, see the following sections:

- Viewing and interpreting system log message descriptions
- Viewing system log message descriptions

Viewing and interpreting system log message descriptions

The messages in this section are available at the time of publishing this IBM Redbooks publication. To view the list of messages that apply to the version of the Junos software that is running on a routing platform, access the Junos CLI operational mode and enter the following command:

```
user@host> help syslog ?
```

To display the list of available descriptions for tags whose names begin with a specific character string, substitute the string in all uppercase letters as shown in the following example. For the variable *TAG-PREFIX* there is no space between the prefix and the question mark.

```
user@host> help syslog TAG-PREFIX?
```

To view the complete descriptions for tags whose name includes a regular expression, substitute the expression for the variable *regex*. The match is not case-sensitive.

```
user@host> help syslog regex
```

To view the complete description of a particular message, substitute its name for the variable TAG (in all uppercase letters):

```
user@host> help syslog TAG
```

Table 13-17 describes the fields in a system log message description in this reference or in the CLI.

Table 13-17 Fields in the system log message descriptions

Field name in the reference	Field name in the CLI	Description
–	Name	The message tag in all capital letters
System Log Message	Message	Text of the message written to the system log. In the log, a specific value is substituted for each variable that is displayed in italics in this reference or in angle brackets (<>) in the CLI. In this reference, the message text is displayed on the second line of the `System Log Message` field. The first line is the message tag, the same text as in the CLI Name field. The prefix on each tag identifies the message source and the rest of the tag indicates the specific event or error. See *JUNOS Software System Log Messages Reference*, GA32-0675, for a listing of the prefixes for which this reference includes entries.
–	Help	Short description of the message, which is also displayed in the right column of the CLI output for the **`help syslog`** command when the output lists multiple messages.
Description	Description	More detailed explanation of the message.
Type	Type	Category to which the message belongs: ► Error: The message reports an error or failure condition that might require corrective action. ► Event: The message reports a condition or occurrence that does not generally require corrective action.
Severity	Severity	Message severity level as described in Table 13-11 on page 477.
Cause	Cause	Optional: Possible cause for message generation. More than one cause is possible.

Viewing system log message descriptions

Example 13-13 shows how to view the list of all currently available system log message descriptions.

Example 13-13 Displaying the syslog message descriptions

```
root> help syslog ?
Possible completions:
  <[Enter]>            Execute this command
  <syslog-tag>         System log tag or regular expression
  ACCT_ACCOUNTING_FERROR  Error occurred during file processing
  ACCT_ACCOUNTING_FOPEN_ERROR  Open operation failed on file
  ACCT_ACCOUNTING_SMALL_FILE_SIZE  Maximum file size is smaller than record size
  ACCT_BAD_RECORD_FORMAT  Record format does not match accounting profile
  ACCT_CU_RTSLIB_ERROR  Error occurred obtaining current class usage statistics
  ACCT_FORK_ERR        Could not create child process
  ACCT_FORK_LIMIT_EXCEEDED  Could not create child process because of limit
  ACCT_GETHOSTNAME_ERROR  gethostname function failed
.
.
.
. <--------------------------Content deleted for brevity
.
.
.
UI_TACPLUS_ERROR     Send to TACACS+ failed
  UI_VERSION_FAILED    mgd could not retrieve version from kernel
  UI_WRITE_RECONNECT   mgd reconnected to peer process
  VRRPD_ADVERT_TIME_MISMATCH  VRRP advertisement timer value was invalid
  VRRPD_AUTH_INFO_INVALID  Authentication information in VRRP packet was invalid
  VRRPD_GET_TRAP_HEADER_FAILED  vrrpd could not obtain SNMP trap header
  VRRPD_LINK_LOCAL_ADD_MISMATCH  Link local address was incorrect
  VRRPD_MISSING_VIP    VRRP packet did not include virtual IP address
  VRRPD_NEW_BACKUP     Interface became VRRP backup
  VRRPD_NEW_MASTER     Interface became VRRP master
  VRRPD_VIP_COUNT_MISMATCH  Number of virtual IP addresses was invalid
  |                    Pipe through a command
```

You can view the list of descriptions for all currently available system log messages for tags that begin with the letters ACCT as shown in Example 13-14. There is no space between ACCT and the question mark, and some descriptions are shortened for legibility.

Example 13-14 Displaying the syslog messages with the ACCT tag

```
root> help syslog ACCT?
Possible completions:
  <syslog-tag>         System log tag or regular expression
  ACCT_ACCOUNTING_FERROR  Error occurred during file processing
  ACCT_ACCOUNTING_FOPEN_ERROR  Open operation failed on file
  ACCT_ACCOUNTING_SMALL_FILE_SIZE  Maximum file size is smaller than record size
  ACCT_BAD_RECORD_FORMAT  Record format does not match accounting profile
  ACCT_CU_RTSLIB_ERROR  Error occurred obtaining current class usage statistics
  ACCT_FORK_ERR        Could not create child process
  ACCT_FORK_LIMIT_EXCEEDED  Could not create child process because of limit
  ACCT_GETHOSTNAME_ERROR  gethostname function failed
  ACCT_MALLOC_FAILURE  Memory allocation failed
```

```
ACCT_UNDEFINED_COUNTER_NAME  Filter profile used undefined counter name
ACCT_XFER_FAILED     Attempt to transfer file failed
ACCT_XFER_POPEN_FAIL  File transfer failed
```

13.8 More information

For more information about all the topics discussed in this chapter, see the following documentation that can be found at the following website:

http://www-947.ibm.com/systems/support/supportsite.wss/brandmain?brandind=5375876

- *JUNOS Software System Basics and Services Command Reference*, GA32-0671-02.
- *JUNOS Software Services Interfaces Configuration Guide,* GA32-0707-02
- *JUNOS Software Network Management Configuration Guide,* GA32-0698
- *JUNOS Software System Log Messages Reference*, GA32-0675
- *JUNOS Software Network Interfaces Command Reference*, GA32-0706
- *JUNOS Software Routing Protocols and Policies Command Reference*, GA32-0673
- *JUNOS Software CLI User Guide*, GA32-0697.

14

Maintenance and analysis

This chapter describes the most important tasks that are required for maintenance and analysis of the IBM j-type series switches and routers.

This chapter includes the following topics:

- Basic systems functions
- Network utilities
- Tools for diagnosing the device
- Managing the Junos software
- Overview of the file system
- Managing the configuration
- Chassis and interfaces alarms

© Copyright IBM Corp. 2011. All rights reserved.

14.1 Basic systems functions

This section explains how to perform basic systems tasks that are required to perform the following tasks:

- Reboot the system.
- Halt the system.
- Manage files.
- Generate a rescue configuration.
- Revert to the rescue configuration.
- Revert to the default factory settings.
- Recover the root password.

14.1.1 Rebooting or halting a system

Example 14-1 shows how to reboot the device by using a command-line interface (CLI) command.

Example 14-1 Rebooting a system

```
ibm@J08E-re0> request system reboot ?
Possible completions:
  <[Enter]>            Execute this command
  at                   Time at which to perform the operation
  in                   Number of minutes to delay before operation
  media                Boot media for next boot
  message              Message to display to all users
  other-routing-engine  Reboot the other Routing Engine
  slice                Partition on boot media to boot from
  |                    Pipe through a command
```

The following options are available:

none To reboot the software immediately.

other-routing-engine Optional: To reboot the other Routing Engine from which the command is issued. For example, if you issue the command from the master Routing Engine, the backup Routing Engine is rebooted. Similarly, if you issue the command from the backup Routing Engine, the master Routing Engine is rebooted.

at *time* Optional: The time at which to reboot the software, which is specified in one of the following ways:

- `now`: To stop or reboot the software immediately. This option is the default setting.
- `+`*`minutes`*: The number of minutes from now to reboot the software.
- *`yymmddhhmm`*: The absolute time at which to reboot the software, specified by the year, month, day, hour, and minute.
- *`hh:mm`*: The absolute time on the current day at which to stop the software, specified in 24-hour time.

in *minutes* Optional: The number of minutes from now to reboot the software. This option is an alias for the `at +`*`minutes`* option.

media (*compact-flash* | *disk* | *removable-compact-flash* | *usb*)
Optional: The boot medium for the next start.

message "*text*" Optional: The message to show to all system users before stopping or rebooting the software.

Reboot requests are recorded in the system log files, which you can view by using the **show log** command. Also, the names of any running processes that are scheduled to be shut down are changed. You can view the process names with the **show system processes** command.

Restarting a router with two Routing Engines: To restart a router that has two Routing Engines, restart the backup Routing Engine (if you have upgraded it) first, and then restart the master Routing Engine.

Example 14-2 shows how to halt the device by using a CLI command.

Example 14-2 Halting a system

```
ibm@J08E-re0> request system halt ?
Possible completions:
  <[Enter]>            Execute this command
  at                   Time at which to perform the operation
  both-routing-engines  Halt both Routing Engines
  in                   Number of minutes to delay before operation
  media                Boot media for next boot
  message              Message to display to all users
  other-routing-engine  Halt other Routing Engine
  slice                Partition on boot media to boot from
  |                    Pipe through a command
```

The following options are available:

at *time* Optional: The time at which to halt the software, which is specified in one of the following ways:

- `now`: To stop the software immediately. This option is the default setting.
- *+minutes*: The number of minutes from now to stop the software.
- *yymmddhhmm*: The absolute time at which to stop the software, specified by the year, month, day, hour, and minute.
- *hh:mm*: The absolute time on the current day at which to stop the software, specified in 24-hour time.

both-routing-engines Optional: To halt both Routing Engines at the same time.

in *minutes* Optional: The number of minutes from now to stop the software. This option is an alias for the `at +minutes` option.

media (compact-flash | disk | removable-compact-flash | usb)
Optional: The boot medium for the next start.

message "text" Optional: The message to show to all system users before stopping the software.

other-routing-engine Optional: To halt the other Routing Engine from which the command is issued. For example, if you enter the command from the master Routing Engine, the backup Routing Engine is halted. Similarly, if you

enter the command from the backup Routing Engine, the master Routing Engine is halted.

14.1.2 Managing files

You can use the J-Web interface or a CLI to perform routine file management operations. Such operations include archiving log files and deleting unused log files, cleaning up temporary files and crash files, and downloading log files from the routing platform to your computer. For an overview of the file system, see 14.5, “Overview of the file system” on page 521.

For example, you can clean up, copy, and delete files in the following ways:

- Cleaning up files

  You can use the CLI **`request system storage cleanup`** command to rotate log files and delete unnecessary files on the services router as shown in Example 14-3. If your system is running low on storage space, the file cleanup procedure quickly identifies files that can be deleted.

  The file cleanup procedure entails running the following tasks:

  – Rotate log files. All information in the current log files is archived, old archives are deleted, and fresh log files are created.

  – Delete log files in the `/var/log` directory. Any files that are not currently being written to are deleted.

  – Delete all crash files in the `/var/crash` directory. Any core files that the device has written during an error are deleted.

  – Delete all software images (TGZ files) in the `/var/sw/pkg` directory. Any software images that are copied to this directory during software upgrades are deleted.

  Example 14-3 Requesting system cleanup

```
ibm@J02M> request system storage cleanup ?
Possible completions:
  <[Enter]>            Execute this command
  dry-run              Only list the cleanup candidates, do not remove them
  |                    Pipe through a command
```

- Copying files

  You can use the **`file copy`** command to copy files from one place to another on the local router or between the local router and a remote system as shown in Example 14-4.

  Example 14-4 Copying files

```
ibm@J11M-re0> file copy day1config ftp://ibm@10.1.1.203
Password for ibm@10.1.1.203:
ftp://ibm@10.1.1.203/day1config               100% of  299  B   16 kBps
```

- Deleting files

  You can use the **file delete** command to delete a file on the local router as shown in Example 14-5.

  Example 14-5 Deleting files

```
ibm@J11M-re0> file delete day1config ?
Possible completions:
  <[Enter]>            Execute this command
  purge                Overwrite regular files before deleting them
  |                    Pipe through a command
```

14.1.3 Setting a rescue configuration

With a *rescue configuration*, you can define a known working configuration or a configuration with a known state that you can roll back to at any time. This type of configuration alleviates the necessity of having to remember the rollback number with the rollback command. You use the rescue configuration when you must roll back to a known configuration. You also use it as a last resort if your router configuration and the backup configuration files become damaged beyond repair.

To set the current active configuration as the rescue configuration, use the following command:

```
ibm@J08E-re0> request system configuration rescue save
```

To delete an existing rescue configuration, use the following command:

```
ibm@J08E-re0> request system configuration rescue delete
```

14.1.4 Reverting to the rescue configuration

The following sections explain how to revert to the rescue configuration.

Using LCD panel

LCD panel: The following procedure is only supported on IBM j-type e-series Ethernet switches.

Someone might inadvertently commit a configuration that denies management access to an IBM j-type e-series Ethernet switch, in which case the console port is not accessible. In this case, you can overwrite the invalid configuration and replace it with the rescue configuration by using the LCD panel on the switch. The rescue configuration is a previously committed, valid configuration.

To revert the switch to the rescue configuration, follow these steps:

1. At the LCD panel on the switch, press Menu until you see `MAINTENANCE MENU`.
2. Press Enter.
3. Press Menu until you see `Load Rescue`.
4. Press Enter.
5. When `Commit Rescue` is displayed, press Enter.

The LCD panel shows the `Commit Rescue in Progress` message. When the reversion is complete, it shows the idle menu.

No rescue configuration: If no rescue configuration is in the device, a failing message is displayed.

Using the CLI

To return to the rescue configuration, use the **`rollback rescue`** configuration mode command:

```
ibm@J11M-re0# rollback rescue
```

To activate the rescue configuration that you have loaded, use the **`commit`** command:

```
ibm@J11M-re0# commit
```

No rescue configuration: If no rescue configuration is in the device, a failing message is displayed.

14.1.5 Reverting to the default factory settings

For any reason, if the current active configuration fails, you can revert to the default factory configuration. The default factory configuration contains the basic configuration settings. It is the first configuration of the switch and is loaded when the switch is first installed and powered on.

Using the LCD panel

Use the LCD panel to revert to the default factory configuration if you want to run EZsetup. When you use the CLI to revert to the default factory configuration, the configuration for the root password is retained, and you cannot run EZSetup.

Important: You might want to convert an IBM Ethernet Switch J48E from a member of a multimember Virtual Chassis configuration to a stand-alone switch. In this case, first disconnect the cables connected to the Virtual Chassis ports (VCPs). By using the Menu button, you delete all modified configuration parameters, including Virtual Chassis parameters such as member ID, mastership priority, and setting VCP uplinks.

To revert to the default factory configuration, follow these steps:

1. Press the Menu button until you see `MAINTENANCE MENU` on the panel.
2. Press the Enter button.
3. Press Menu until you see `FACTORY DEFAULT`.
4. Press Enter.
5. When the display says `RESTORE DEFAULT?`, press Enter.

The screen shows a `FACTORY DEFAULT IN PROGRESS` message and returns to the idle menu.

Using the CLI

The following command replaces the current active configuration with the default factory configuration:

```
ibm@J11M-re0# load factory-default
```

Important: The `load factory default` command by itself is not supported on IBM Ethernet Switch J48E configured in a Virtual Chassis with multiple members. In a multimember Virtual Chassis configuration, you can revert to the default factory configuration while retaining the Virtual Chassis parameters (member ID, mastership priority, or settings of VCP uplinks) by entering the following commands in the order shown:

```
user@switch# load factory default
user@switch# delete system commit factory-settings
user@switch# commit
```

14.1.6 Recovering the root password

To recover the root password for the IBM j-type e-series Ethernet switches and IBM j-type m-series Ethernet routers, follow these steps:

1. Power off your switch or router:
   - For an IBM j-type e-series Ethernet switch, power off your switch by unplugging the power cord or turning off the power at the wall switch.
   - For an IBM j-type m-series Ethernet router, power off the router by pressing the power button on the front panel.
2. Connect the console port to the management PC by using the serial cable that is supplied with the device.
3. Power on your switch or router:
   - For an IBM j-type e-series Ethernet switch, power on your switch by plugging in the power cord or turning on the power at the wall switch.
   - For an IBM j-type m-series Ethernet router, power on the router by pressing the power button on the front panel. Verify that the POWER LED on the front panel turns green.
4. When the following message is displayed, press the Spacebar to access the bootstrap loader command prompt of the switch:

   ```
   Hit [Enter] to boot immediately, or space bar for command prompt. Booting
   [kernel] in 1 second...
   ```
5. At the `loader` prompt, type the **`boot -s`** command to start the system in single-user mode as in the following example:

   ```
   loader> boot -s
   ```
6. At the following prompt, type **`recovery`** to start the root password recovery procedure:

   ```
   Enter full pathname of shell or 'recovery' for root password recovery or RETURN
   for /bin/sh: recovery
   ```

   A series of messages describes consistency checks, mounting of file systems, and initialization and checkout of management services. Then the CLI prompt is displayed.
7. Enter configuration mode in the CLI:

   ```
   user@switch> configure
   ```
8. Set the root password as shown in the following example:

   ```
   user@switch# set system root-authentication plain-text-password
   ```
9. At the following prompt, enter the new root password:

   ```
   New password: ibm123
   ```

10. At the second prompt, re-enter the new root password:

```
Retype new password:
```

11. If you are finished configuring the network, commit the configuration:

```
root@switch# commit
```

12. Exit the configuration mode in the CLI:

```
root@switch# exit
```

13. Exit the operational mode in the CLI.

```
root@switch> exit
```

14. At the prompt, enter y to reboot the switch:

```
Reboot the system? [y/n] y
```

14.2 Network utilities

This section describes the basic network utilities that are commonly used for verification and troubleshooting.

14.2.1 The ping and traceroute commands

You can use the **ping** and **traceroute** commands to determine network reachability and to determine the path that packets take to reach a destination. Example 14-6 shows the options that are available for these commands. Use Ctrl+C to stop **ping** and **traceroute** commands.

Example 14-6 The Ping and traceroute commands

```
ibm@J11M-re0> ping 10.1.1.133 ?
Possible completions:
  <[Enter]>            Execute this command
  bypass-routing       Bypass routing table, use specified interface
  count                Number of ping requests to send (1..2000000000 packets)
  detail               Display incoming interface of received packet
  do-not-fragment      Don't fragment echo request packets (IPv4)
  inet                 Force ping to IPv4 destination
  inet6                Force ping to IPv6 destination
  interface            Source interface (multicast, all-ones, unrouted packets)
  interval             Delay between ping requests (seconds)
  logical-router       Name of logical router
+ loose-source         Intermediate loose source route entry (IPv4)
  no-resolve           Don't attempt to print addresses symbolically
  pattern              Hexadecimal fill pattern
  rapid                Send requests rapidly (default count of 5)
  record-route         Record and report packet's path (IPv4)
  routing-instance     Routing instance for ping attempt
  size                 Size of request packets (0..65468 bytes)
  source               Source address of echo request
  strict               Use strict source route option (IPv4)
+ strict-source        Intermediate strict source route entry (IPv4)
  tos                  IP type-of-service value (0..255)
  ttl                  IP time-to-live value (IPv6 hop-limit value) (0..255 hops)
  verbose              Display detailed output
  wait                 Delay after sending last packet (seconds)
```

```
  |                     Pipe through a command

bm@J11M-re0> ping 10.1.1.133
PING 10.1.1.133 (10.1.1.133): 56 data bytes
64 bytes from 10.1.1.133: icmp_seq=0 ttl=128 time=0.802 ms
64 bytes from 10.1.1.133: icmp_seq=1 ttl=128 time=0.235 ms
64 bytes from 10.1.1.133: icmp_seq=2 ttl=128 time=0.234 ms
^C
--- 10.1.1.133 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.234/0.424/0.802/0.268 ms


ibm@J11M-re0> traceroute ?
Possible completions:
  <host>               Hostname or address of remote host
  as-number-lookup     Look up AS numbers for each hop
  bypass-routing       Bypass routing table, use specified interface
  ethernet             Trace route to an ethernet host by unicast mac address
  gateway              Address of router gateway to route through
  inet                 Force traceroute to IPv4 destination
  inet6                Force traceroute to IPv6 destination
  interface            Name of interface to use for outgoing traffic
  logical-router       Name of logical router
  monitor              Monitor network connection to remote host
  mpls                 Trace MPLS paths
  no-resolve           Don't attempt to print addresses symbolically
  routing-instance     Name of routing instance for traceroute attempt
  source               Source address to use in outgoing traceroute packets
  tos                  IP type-of-service field (IPv4) (0..255)
  ttl                  IP maximum time-to-live value (or IPv6 maximum hop-limit
value)
  wait                 Number of seconds to wait for response (seconds)


ibm@J11M-re0> traceroute 10.1.1.133
traceroute to 10.1.1.133 (10.1.1.133), 30 hops max, 40 byte packets
 1  * * *
```

14.2.2 Port mirroring on IBM j-type e-series Ethernet switches

Port mirroring is needed for traffic analysis on a switch because a switch, unlike a hub, does not broadcast packets to every port on the device. The switch sends packets only to the port to which the destination device is connected. You configure port mirroring on the switch to send copies of unicast traffic to either a local analyzer port or an analyzer virtual local area network (VLAN). Then you can analyze the mirrored traffic by using a protocol analyzer application. The protocol analyzer application can run on either a computer connected to the analyzer output interface or a remote monitoring station.

Use port mirroring to facilitate traffic analysis on your IBM j-type e-series Ethernet switches on a packet level. Use port mirroring as part of monitoring traffic for the following reasons among others:

- To enforce policies concerning network usage and file sharing
- To identify sources of problems on your network by locating abnormal or heavy bandwidth usage from particular stations or applications

Disable port mirroring when you are not using it and select specific interfaces as input to the port mirror analyzer in preference to using the `all` keyword option. You can also limit the amount of mirrored traffic by using statistical sampling, setting a ratio to select a statistical sample, or using a firewall filter. Mirroring only the necessary packets reduces any potential affect on performance.

With local port mirroring, traffic from multiple ports is replicated to the analyzer output interface. If the output interface for an analyzer reaches capacity, packets are dropped. Consider whether the traffic that is being mirrored exceeds the capacity of the analyzer output interface.

Port mirroring copies packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use port mirroring to copy these packets:

- Packets entering or exiting a port

  You can mirror the packets in any combination (up to 256 ports). For example, you can send copies of the packets entering some ports and the packets exiting other ports to the same local analyzer port or analyzer VLAN.
- Packets entering a VLAN on IBM Ethernet Switch J48E

  You can mirror the packets entering a VLAN on this switch to either a local analyzer port or to an analyzer VLAN. You can configure multiple VLANs (up to 256 VLANs), including a VLAN range and private VLANs (PVLANs), as ingress input to an analyzer.
- Packets exiting a VLAN on IBM Ethernet Switch J08E or J16E

  You can mirror the packets exiting a VLAN to either a local analyzer port or to an analyzer VLAN. You can configure multiple VLANs (up to 256 VLANs), including a VLAN range and PVLANs, as egress input to an analyzer.

Limitations of port mirroring

Port mirroring has the following limitations:

- On an IBM Ethernet Switch J48E, you can enable only one analyzer (port mirroring configuration).
- On an IBM Ethernet Switch J08E or J16E, you can enable a maximum of seven analyzers (port mirroring configurations).
- Packets with physical layer errors are filtered out and thus are not sent to the analyzer port or analyzer VLAN.
- You cannot mirror packets exiting or entering the following ports:
  - Dedicated Virtual Chassis ports (VCPs)
  - Management port (me0 or vme0)
  - Routed VLAN interfaces (RVIs)
- On IBM Ethernet Switch J08E or J16E, you can set a ratio only for ingress packets.

- On IBM Ethernet Switch J48E, mirrored packets that are exiting a tagged interface might contain an incorrect VLAN ID.
- On IBM Ethernet Switch J48E, tagged packets that are mirrored to an analyzer port might contain an incorrect Ethertype.

Terminology

Port mirroring involves the following terminology:

Analyzer A port-mirroring configuration on an EX Series switch.

Analyzer output interface or monitor port
Interface to which mirrored traffic is sent and to which a protocol analyzer application is connected.

Analyzer VLAN or monitor VLAN
A VLAN to which mirrored traffic is sent. The mirrored traffic can be used by a protocol analyzer application. The monitor VLAN is spread across the switches in your network.

Firewall-based Analyzer
An analyzer session that has only an “output” stanza. A firewall-based Analyzer must be used along with a firewall to achieve the functionality of an analyzer.

Input interface or mirrored port
An interface on the switch that is being mirrored, on traffic either entering or exiting the interface.

Monitoring station A computer running a protocol analyzer application.

Native analyzer session
An analyzer session that has both “input” and “output” stanzas.

Policy-based mirroring
The mirroring of packets that match the match items in the defined firewall filter term. The action item analyzer `analyzer-name` is used in the firewall filter to send the packets to the port mirror analyzer.

Protocol analyzer application
An application that is used to examine packets that are transmitted across a network segment. Also commonly called *network analyzer*, *packet sniffer*, or *probe*.

Remote port mirroring
Functions the same as local port mirroring, except that the mirrored traffic is not copied to a local analyzer port. Instead, it is flooded into an analyzer VLAN that you create specifically for receiving mirrored traffic.

Statistical sampling The configuration of the system to mirror a sampling of the packets, by setting a ratio of 1:*x*, where *x* is a value in the range 1–2047. For example, when the ratio is set to 1, all packets are copied to the analyzer. When the ratio is set to 200, 1 of every 200 packets is copied.

Configuring port mirroring for local traffic

To mirror interface or VLAN traffic to a port on the same switch, follow these steps:

1. Select a name for the port mirroring configuration (TEST in this case) and specify the input (interface ge-1/0/46.0 in this case):

   ```
   {master}[edit ethernet-switching-options]
   ibm@J08E-re0# set analyzer TEST input ingress interface ge-1/0/46.0
   ```

2. Optional: Specify a statistical sampling of the packets by setting a ratio:

   ```
   {master}[edit ethernet-switching-options]
   ibm@J08E-re0# set analyzer TEST ratio 10
   ```

   When the ratio is set to 10, 1 of every 10 packets is mirrored to the analyzer. You can use statistical sampling to reduce the volume of mirrored traffic, because a high volume of mirrored traffic can be performance intensive for the switch.

3. Configure the destination interface for the mirrored packets:

   ```
   {master}[edit ethernet-switching-options]
   ibm@J08E-re0# set analyzer TEST output interface ge-1/0/47.0
   ```

Configuring port mirroring for remote traffic

To mirror traffic that is traversing interfaces or a VLAN on the switch to a VLAN for analysis from a remote location, follow these steps:

1. Configure a VLAN to carry the mirrored traffic (VLAN 1000 in this case):

   ```
   {master}[edit]
   ibm@J08E-re0# set vlans ANALYZER vlan-id 1000
   ```

2. Associate the uplink trunk port with the remote analyzer VLAN:

   ```
   {master}[edit]
   ibm@J08E-re0# set interfaces ge-1/0/45 unit 0 family ethernet-switching
   port-mode trunk vlan members ANALYZER
   ```

3. Configure the analyzer:

   a. Select a name for the port mirroring configuration (REMOTE-MONITOR in this case). Always set loss priority high when configuring for remote port mirroring:

      ```
      {master}[edit ethernet-switching-options]
      ibm@J08E-re0# set analyzer REMOTE-MONITOR loss-priority high
      ```

   b. Specify the traffic to be mirrored (interface ge0-1/0/46 in this case):

      ```
      {master}[edit ethernet-switching-options]
      ibm@J08E-re0# set analyzer REMOTE-MONITOR input ingress interface ge-1/0/46
      ```

   c. Specify the corresponding VLAN as the output:

      ```
      {master}[edit ethernet-switching-options]
      ibm@J08E-re0# set analyzer REMOTE-MONITOR output vlan ANALYZER
      ```

4. Optional: Configure the sampling rate:

   ```
   {master}[edit ethernet-switching-options]
   ibm@J08E-re0# set analyzer REMOTE-MONITOR ratio 20
   ```

For more information about IBM j-type e-series port mirroring, see “Configuring Port Mirroring to Analyze Traffic (CLI Procedure)” at the following address:

http://www.juniper.net/techpubs/en_US/junos10.1/topics/task/configuration/port-mirroring-cli.html

14.2.3 Port mirroring on IBM j-type m-series Ethernet routers

On routing platforms that contain an Internet Processor II ASIC, you can send a copy of any incoming packet from the routing platform to an external host address or a packet analyzer for analysis. This concept is known as *port mirroring*. This section explains port mirroring on IBM j-type m-series Ethernet routers.

Overview of Layer 2 mirroring

In Junos Release 9.3 and later, IBM j-type m-series Ethernet router support port mirroring for Layer 2 bridging traffic. With Layer 2 port mirroring, you can specify the manner in which incoming and outgoing packets at specified ports in a bridging environment are monitored and sampled. You can also specify the manner in which copies of the sampled packet are forwarded to another destination and where the packets can be analyzed.

In a Layer 2 environment, IBM j-type m-series Ethernet routers support port mirroring of VPLS (family bridge or family VPLS) traffic. IBM j-type m-series Ethernet routers also support port mirroring for Layer 2 VPNs.

For more information about Layer 2 mirroring and configuration guidance, see the *JUNOS Software IBM j-type m-series Ethernet Routers Layer 2 Configuration Guide,* GA32-0708.

Overview of Layer 3 mirroring

Layer 3 port mirroring is the ability of a router to send a copy of an IPv4 or IPv6 packet to an external host address or a packet analyzer for analysis. Port mirroring is different from traffic sampling. In traffic sampling, a sampling key based on the packet headers is sent to the Routing Engine. There, the key can be placed in a file, or **cflowd** packets based on the key can be sent to a **cflowd** server. In port mirroring, the entire packet is copied and sent through a next-hop interface.

One application for port mirroring sends a duplicate packet to a virtual tunnel. A next-hop group can then be configured to forward copies of this duplicate packet to several interfaces. All IBM j-type m-series Ethernet routers support IPv4 and IPv6 mirroring and VPLS traffic mirroring.

For more information about Layer 3 mirroring and configuration guidance, see the *JUNOS Software Policy Framework Configuration Guide,* GA32-0704.

14.2.4 Telnet, SSH, and FTP clients

The CLI supports Telnet, Secure Shell (SSH), and FTP clients. Example 14-7 shows the available arguments for these commands.

Example 14-7 Telnet, SSH, and FTP

```
ibm@J08E-re0> telnet ?
Possible completions:
  <host>               Hostname or address or remote host
  8bit                 Use 8-bit data path
  bypass-routing       Bypass routing table, use specified interface
  inet                 Force telnet to IPv4 destination
  inet6                Force telnet to IPv6 destination
  interface            Name of interface for outgoing traffic
  no-resolve           Don't attempt to print addresses symbolically
  port                 Port number or service name on remote host
  routing-instance     Name of routing instance for telnet session
```

```
  source                Source address to use in telnet connection

ibm@J08E-re0> ssh ?
Possible completions:
  <host>                Hostname or address of remote host
  bypass-routing        Bypass routing table, use specified interface
  inet                  Force ssh to IPv4 destination
  inet6                 Force ssh to IPv6 destination
  interface             Name of interface for outgoing traffic
  routing-instance      Name of routing instance for ssh session
  source                Source address for ssh session
  v1                    Force ssh to try protocol version 1 only
  v2                    Force ssh to try protocol version 2 only

ibm@J08E-re0> ftp ?
Possible completions:
  <host>                Hostname or address of remote host
  bypass-routing        Bypass routing table, use specified interface
  inet                  Force FTP to IPv4 destination
  inet6                 Force FTP to IPv6 destination
  interface             Name of interface for outgoing traffic
  routing-instance      Name of routing instance for FTP session
  source                Source address for FTP session
```

14.3 Tools for diagnosing the device

You can diagnose the device with CLI operational mode commands. CLI command output is displayed on the screen of your console or management device, or you can filter the output to a file.

This section explains the following commands and files:

- The show interfaces command
- The monitor interface command
- The monitor traffic command
- Monitoring system commands
- Viewing log and trace files

14.3.1 The show interfaces command

You can enter the following **show** commands in the CLI to view interface status and traffic statistics:

- **show interfaces terse**
- **show interfaces** *interface-name*
- **show interfaces extensive** *interface-name*

The **show interfaces terse** command (Example 14-8) shows summary information about interfaces.

Example 14-8 The show interfaces terse command

```
ibm@J02M> show interfaces terse
Interface               Admin Link Proto    Local                 Remote
ge-1/0/0                up    down
lc-1/0/0                up    up
lc-1/0/0.32769          up    up   vpls
ge-1/0/1                up    up
ge-1/0/1.100            up    up   bridge
ge-1/0/1.200            up    up   bridge
ge-1/0/1.32767          up    up   multiservice
ge-1/0/2                up    up
ge-1/0/2.101            up    up   bridge
ge-1/0/2.201            up    up   bridge
ge-1/0/2.32767          up    up   multiservice
ge-1/0/3                up    down
ge-1/0/3.0              up    down bridge
ge-1/0/4                up    up
ge-1/0/4.100            up    up   bridge
ge-1/0/4.101            up    up   bridge
ge-1/0/4.200            up    up   bridge
ge-1/0/4.201            up    up   bridge
ge-1/0/4.32767          up    up   multiservice


####(output omitted for brevity)###
```

Table 14-1 shows the output fields of the **show interfaces terse** command.

Table 14-1 The output fields for the show interfaces terse command

Field name	Description
Interface	Interface name.
Admin	Whether the interface is turned on (up) or off (down).
Link	Link state: up or down.
Proto	Protocol family configured on the logical interface. A logical interface on a router that supports Ethernet object access method (OAM) always shows the multiservice protocol.
Local	Local IP address of the logical interface.
Remote	Remote IP address of the logical interface.

The **show interfaces** *interface-name* command (Example 14-9) shows standard information about the specified interface.

Example 14-9 The show interfaces command

```
ibm@J02M> show interfaces ge-1/0/1
Physical interface: ge-1/0/1, Enabled, Physical link is Up
  Interface index: 134, SNMP ifIndex: 507
  Link-level type: 52, MTU: 1522, Speed: 1000mbps, BPDU Error: None,
  MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled,
  Flow control: Enabled, Auto-negotiation: Enabled, Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  CoS queues     : 8 supported, 8 maximum usable queues
  Current address: 00:17:cb:db:ea:96, Hardware address: 00:17:cb:db:ea:96
  Last flapped   : 2010-05-24 18:49:34 UTC (23:44:20 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 856 bps (1 pps)
  Active alarms  : None
  Active defects : None

  Logical interface ge-1/0/1.100 (Index 80) (SNMP ifIndex 569)
    Flags: SNMP-Traps 0x20004000 VLAN-Tag [ 0x8100.100 ]
    Encapsulation: VLAN-Bridge
    Input packets : 103
    Output packets: 607277
    Protocol bridge, MTU: 1522

  Logical interface ge-1/0/1.200 (Index 69) (SNMP ifIndex 567)
    Flags: SNMP-Traps 0x20004000 VLAN-Tag [ 0x8100.200 ]
    Encapsulation: VLAN-Bridge
    Input packets : 26
    Output packets: 259471181
    Protocol bridge, MTU: 1522

  Logical interface ge-1/0/1.32767 (Index 77) (SNMP ifIndex 565)
    Flags: SNMP-Traps 0x4004000 VLAN-Tag [ 0x0000.0 ]  Encapsulation: ENET2
    Input packets : 0
    Output packets: 13605
    Protocol multiservice, MTU: Unlimited
      Flags: None
```

The **show interfaces extensive** command shows all possible information about the interfaces that are installed in the device. You can specify a particular interface as shown in Example 14-10.

Example 14-10 The show interface extensive command

```
ibm@J02M> show interfaces extensive ge-1/0/0
Physical interface: ge-1/0/0, Enabled, Physical link is Down
  Interface index: 133, SNMP ifIndex: 506, Generation: 136
  Link-level type: Ethernet, MTU: 1514, Speed: 1000mbps, BPDU Error: None,
  MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled,
  Flow control: Enabled, Auto-negotiation: Enabled, Remote fault: Online
  Device flags   : Present Running Down
  Interface flags: Hardware-Down SNMP-Traps Internal: 0x4000
```

```
Link flags     : None
CoS queues     : 8 supported, 8 maximum usable queues
Hold-times     : Up 0 ms, Down 0 ms
Current address: 00:17:cb:db:ea:95, Hardware address: 00:17:cb:db:ea:95
Last flapped   : 2010-05-20 21:35:28 UTC (4d 21:04 ago)
Statistics last cleared: Never
Traffic statistics:
 Input  bytes  :             55214694                    0 bps
 Output bytes  :             75826407                    0 bps
 Input  packets:               394090                    0 pps
 Output packets:               494379                    0 pps
 IPv6 transit statistics:
  Input  bytes  :                   0
  Output bytes  :                   0
  Input  packets:                   0
  Output packets:                   0
Dropped traffic statistics due to STP State:
 Input  bytes  :                    0
 Output bytes  :                    0
 Input  packets:                    0
 Output packets:                    0
Input errors:
  Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0,
  L3 incompletes: 0, L2 channel errors: 0, L2 mismatch timeouts: 0,
  FIFO errors: 0, Resource errors: 0
Output errors:
  Carrier transitions: 6, Errors: 0, Drops: 142, Collisions: 0,
  Aged packets: 0, FIFO errors: 0, HS link CRC errors: 0, MTU errors: 0,
  Resource errors: 0
Egress queues: 8 supported, 4 in use
Queue counters:       Queued packets  Transmitted packets      Dropped packets
  0 best-effort            166002974            166002832                  142
  1 expedited-fo                   0                    0                    0
  2 assured-forw                   0                    0                    0
  3 network-cont              213776               213776                    0
Active alarms  : LINK
Active defects : LINK
MAC statistics:                      Receive         Transmit
  Total octets                      63497653         71944692
  Total packets                       394197           495044
  Unicast packets                     279496           279405
  Broadcast packets                      842             1300
  Multicast packets                   113859           214339
  CRC/Align errors                         0                0
  FIFO errors                              0                0
  MAC control frames                       0              666
  MAC pause frames                         0              666
  Oversized frames                         0
  Jabber frames                            0
  Fragment frames                          0
  VLAN tagged frames                       0
  Code violations                          0
Filter statistics:
  Input packet count                  394197
  Input packet rejects                   107
```

```
    Input DA rejects                        88
    Input SA rejects                         0
    Output packet count                                    495044
    Output packet pad count                                     0
    Output packet error count                                   0
    CAM destination filters: 0, CAM source filters: 0
  Autonegotiation information:
    Negotiation status: Incomplete
  Packet Forwarding Engine configuration:
    Destination slot: 1
  CoS information:
    Direction : Output
    CoS transmit queue               Bandwidth               Buffer Priority   L
imit
                              %            bps     %           usec
    0 best-effort            95      950000000    95              0      low
none
    3 network-control         5       50000000     5              0      low
none
```

14.3.2 The monitor interface command

Use the CLI **monitor interface** command (Example 14-11) to see real-time traffic, error, alarm, and filter statistics about a physical or logical interface.

Example 14-11 The monitor interface command

```
ibm@J02M> monitor interface ge-1/0/1

J02M                              Seconds: 4                   Time: 18:43:00
                                                            Delay: 6/0/6
Interface: ge-1/0/1, Enabled, Link is Up
Encapsulation: Flexible-Ethernet-Services, Speed: 1000mbps
Traffic statistics:                                            Current delta
  Input bytes:                     9526 (0 bps)                        [0]
  Output bytes:             23834803183 (600 bps)                    [436]
  Input packets:                    143 (0 pps)                        [0]
  Output packets:             260094714 (1 pps)                        [6]
Error statistics:
  Input errors:                       0                                [0]
  Input drops:                        0                                [0]
  Input framing errors:               0                                [0]
  Policed discards:                   0                                [0]
  L3 incompletes:                     0                                [0]
  L2 channel errors:                  0                                [0]
  L2 mismatch timeouts:               0                                [0]
  Carrier transitions:               13                                [0]
  Output errors:                      0                                [0]
  Output drops:                       0                                [0]
  Aged packets:                       0                                [0]
Active alarms : None
Active defects: None
Input MAC/Filter statistics:
  Unicast packets                    97                                [0]
  Broadcast packets                  21                                [0]
```

```
  Multicast packets                  32                                  [0]
  Oversized frames                    0                                  [0]
  Packet reject count                 7                                  [0]
  DA rejects                          0                                  [0]
  SA rejects                          0                                  [0]
Output MAC/Filter Statistics:
  Unicast packets                 22094                                  [0]
  Broadcast packets            18362054                                  [0]
  Multicast packets           241710898                                  [5]
  Packet pad count                    0                                  [0]
  Packet error count                  0                                  [0]



Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'
```

The real-time statistics are updated every second. The Current delta and Delta columns show the amount that the statistics counters have changed since the **monitor interface** command was entered or since you cleared the delta counters.

Table 14-2 lists the keys that you use to control the display.

Table 14-2 Options for the monitor interface command

Key	Description
n	Shows information about the next interface. The device scrolls through the physical and logical interfaces in the same order in which they are displayed by the **show interfaces terse** command
q or ESC	Quits the command and returns to the command prompt.
f	Freezes the display, halting the update of the statistics and delta counters.
t	Thaws the display, resuming the update of the statistics and delta counters.
c	Clears (returns to 0) the delta counters in the Current delta column. The statistics counters are not cleared.
i	Displays information about a different interface. You are prompted for the name of a specific interface.

14.3.3 The monitor traffic command

Use the CLI **monitor traffic** command (Example 14-12 on page 510) to display packet headers that are transmitted through network interfaces.

Attention: Using the **monitor traffic** command can degrade system performance. Use filtering options, such as **count** and **matching**, to minimize the affect on packet throughput on the system.

Example 14-12 The monitor traffic command

```
ibm@J02M> monitor traffic ?
Possible completions:
  <[Enter]>            Execute this command
  absolute-sequence    Display absolute TCP sequence numbers
  brief                Display brief output
  count                Number of packets to receive (0..1000000 packets)
  detail               Display detailed output
  extensive            Display extensive output
  interface            Name of interface
  layer2-headers       Display link-level header on each dump line
  matching             Expression for headers of receive packets to match
  no-domain-names      Don't display domain portion of hostnames
  no-promiscuous       Don't put interface into promiscuous mode
  no-resolve           Don't attempt to print addresses symbolically
  no-timestamp         Don't print timestamp on each dump line
  print-ascii          Display packets in ASCII when displaying in hexadecimal f
ormat
  print-hex            Display packets in hexadecimal format
  resolve-timeout      Period of time to wait for each name resolution (seconds)
  size                 Amount of each packet to receive (bytes)
  |                    Pipe through a command
```

To limit the packet header information that is displayed by the **monitor traffic** command, include the **matching** *expression* option.

For more information about monitor traffic command, see “monitor traffic” at the following web address:

http://www.juniper.net/techpubs/en_US/junos10.1/information-products/topic-collections/swcmdref-basics-services/topic-19319.html

Example 14-13 shows how to use the **monitor traffic** command to match TCP traffic.

Example 14-13 Monitoring traffic

```
ibm@J02M> monitor traffic matching tcp
verbose output suppressed, use <detail> or <extensive> for full protocol decode
Address resolution is ON. Use <no-resolve> to avoid any reverse lookup delay.
Address resolution timeout is 4s.
Listening on fxp0, capture size 96 bytes

Reverse lookup for 10.1.1.50 failed (check DNS reachability).
Other reverse lookup failures will not be reported.
Use <no-resolve> to avoid reverse lookups on IP addresses.

18:48:29.902777  In IP 10.1.1.8.stone-design-1 > 10.1.1.50.telnet: . ack 7304593
71 win 64174
18:48:29.902830 Out IP truncated-ip - 221 bytes missing! 10.1.1.50.telnet > 10.1
.1.8.stone-design-1: P 1:242(241) ack 0 win 65535
18:48:30.121533  In IP 10.1.1.8.stone-design-1 > 10.1.1.50.telnet: . ack 242 win
 65535
18:48:30.733351 Out IP truncated-ip - 471 bytes missing! 10.1.1.50.telnet > 10.1
.1.8.stone-design-1: P 242:733(491) ack 0 win 65535
18:48:30.887198  In IP 10.1.1.8.stone-design-1 > 10.1.1.50.telnet: . ack 733 win
 65044
```

```
18:48:31.732763 Out IP truncated-ip - 201 bytes missing! 10.1.1.50.telnet > 10.1
.1.8.stone-design-1: P 733:954(221) ack 0 win 65535
18:48:31.871633  In IP 10.1.1.8.stone-design-1 > 10.1.1.50.telnet: . ack 954 win
 64823
18:48:32.732602 Out IP truncated-ip - 201 bytes missing! 10.1.1.50.telnet > 10.1
.1.8.stone-design-1: P 954:1175(221) ack 0 win 65535
18:48:32.856050  In IP 10.1.1.8.stone-design-1 > 10.1.1.50.telnet: . ack 1175 wi
n 64602
^C
14 packets received by filter
0 packets dropped by kernel
```

14.3.4 Monitoring system commands

You can view system properties by entering the following **show** commands in the CLI configuration editor as shown in Example 14-14:

show system uptime Use this command to show time since system and processes started.

show system user Use this command to show users who are currently logged in.

show system storage Use this command to show local storage data.

show version Use this command to show software process revision levels.

show chassis hardware Use this command to show installed hardware components.

Example 14-14 The show system commands

```
ibm@J02M> show system uptime
Current time: 2010-05-25 18:50:15 UTC
System booted: 2010-05-10 18:04:36 UTC (2w1d 00:45 ago)
Protocols started: 2010-05-10 18:05:40 UTC (2w1d 00:44 ago)
Last configured: 2010-05-22 03:47:46 UTC (3d 15:02 ago) by ibm
 6:50PM  up 15 days, 46 mins, 4 users, load averages: 0.06, 0.03, 0.00

ibm@J02M> show system users
 6:50PM  up 15 days, 46 mins, 4 users, load averages: 0.05, 0.03, 0.00
USER     TTY      FROM                              LOGIN@  IDLE WHAT
ibm      d0       -                                12May10 12days -cli (cli)
ibm      p0       10.1.1.8                         Sat02PM  4:29 -cli (cli)
ibm      p1       10.1.1.8                         1:55PM   2:48 -cli (cli)
ibm      p2       10.1.1.8                         6:24PM      - -cli (cli)

ibm@J02M> show system storage
Filesystem             Size       Used      Avail  Capacity   Mounted on
/dev/ad0s1a            885M       191M       623M       23%  /
devfs                  1.0K       1.0K         0B      100%  /dev
/dev/md0                34M        34M         0B      100%  /packages/mnt/jbas
e
/dev/md1               248M       248M         0B      100%  /packages/mnt/jker
nel-10.1R1.8
/dev/md2                45M        45M         0B      100%  /packages/mnt/jpfe
-X960-10.1R1.8
/dev/md3               5.4M       5.4M         0B      100%  /packages/mnt/jdoc
s-10.1R1.8
/dev/md4                66M        66M         0B      100%  /packages/mnt/jrou
```

```
te-10.1R1.8
/dev/md5                 15M        15M         0B      100%  /packages/mnt/jcry
pto-10.1R1.8
/dev/md6                 36M        36M         0B      100%  /packages/mnt/jpfe
-common-10.1R1.8
/dev/md7                2.0G      10.0K       1.8G        0%  /tmp
/dev/md8                2.0G       1.7M       1.8G        0%  /mfs
/dev/ad0s1e              98M        16K        90M        0%  /config
procfs                  4.0K       4.0K         0B      100%  /proc
/dev/ad2s1f              34G       757M        30G        2%  /var

ibm@J02M> show version
Hostname: J02M
Model: mx240
JUNOS Base OS boot [10.1R1.8]
JUNOS Base OS Software Suite [10.1R1.8]
JUNOS Kernel Software Suite [10.1R1.8]
JUNOS Crypto Software Suite [10.1R1.8]
JUNOS Packet Forwarding Engine Support (M/T Common) [10.1R1.8]
JUNOS Packet Forwarding Engine Support (MX Common) [10.1R1.8]
JUNOS Online Documentation [10.1R1.8]
JUNOS Voice Services Container package [10.1R1.8]
JUNOS Border Gateway Function package [10.1R1.8]
JUNOS Services AACL Container package [10.1R1.8]
JUNOS Services LL-PDF Container package [10.1R1.8]
JUNOS Services Stateful Firewall [10.1R1.8]
JUNOS AppId Services [10.1R1.8]
JUNOS IDP Services [10.1R1.8]
JUNOS Routing Software Suite [10.1R1.8]
```

14.3.5 Viewing log and trace files

You can enter the **monitor start** command to view real-time additions to system logs and trace files:

```
ibm@J02M> monitor start filename
```

When the device adds a record to the file specified by *filename*, the record is shown on the panel. For example, if you configured a system log file named `system-log` (by including the `syslog` statement at the `[edit system]` hierarchy level), you can enter the **monitor start system-log** command to show the records that are added to the system log.

To see a list of files that are being monitored, enter the **monitor list** command. To stop the display of records for a specified file, enter the **monitor stop** *filename* command. For more information about configuring syslog, see Chapter 13, “System management” on page 409.

14.4 Managing the Junos software

To upgrade Junos software on an IBM j-type series switch, you must copy a software package to your device or another system on your local network. Then use either the J-Web interface or the CLI to install the new software package on the device. Finally, you restart the device, which starts from the upgraded software.

After a successful upgrade, back up the new current configuration to a secondary device. During a successful upgrade, the upgrade package removes all files from the `/var/tmp` directory and completely re-installs the existing software. It retains configuration files and similar information, such as SSH and host keys, from the previous version. The previous software package is preserved in a separate disk partition. You can manually revert to it if necessary. If the software installation fails for any reason, such as a loss of power during the installation process, the system returns to the original active installation when you restart.

14.4.1 Software packaging

A software package name is in the following format:

package-name-m.nZx.y-domestic-signed.tgz

where:

package-name	is the name of the package, for example jinstall-ex-4200.
m.n	is the software release, with *m* representing the major release number and *n* representing the minor release number, for example 9.5.
Z	indicates the type of software release, where *R* indicates released software and *B* indicates beta-level software.
x.y	represents the version of the major software release (x) and an internal tracking number (y), for example 1.6.
domestic-signed	for most Junos packages, is used for the United States and Canada.
export-signed	is used for worldwide distribution.

The `jinstall-ex-4200-9.5R1.6-domestic-signed.tgz` file is an example of Junos software package.

14.4.2 System snapshot

You can create copies of the software by using the system snapshot feature. The system snapshot feature takes a “snapshot” of the files that are currently used to run the device and copies all of these files into an alternate memory source. You can then use this snapshot to start the device at the next startup or as a backup start option.

Use the command shown in Example 14-15 to run the snapshot feature.

Example 14-15 System snapshot

```
ibm@J08E-re0> request system snapshot ?
Possible completions:
  <[Enter]>            Execute this command
  as-primary           Setup snapshot to be used in the primary boot device
  media                Media to snapshot to
  partition            Partition the media
  re0                  Archive data and executable areas of RE0
  re1                  Archive data and executable areas of RE1
  routing-engine       Archive data and executable areas of specific Routing
Engine
  slice                Write snapshot to specified partition
  |                    Pipe through a command
```

Backing up IBM j-type e-series Ethernet switches

The snapshot feature copies the complete contents of the `/config` and `/var` directories. The content of these directories includes the running Junos software, the active configuration, and the rescue configuration into an alternate memory source (internal flash, or an external USB flash).

System snapshots on IBM j-type e-series Ethernet switches have the following limitations:

- You can only use snapshots to move files to external memory if the switch was started from internal memory or to move files to internal memory if the switch was started from external memory. You cannot create a snapshot in the memory source that started the switch even if the snapshot is being created on a different partition in the same memory source.
- Snapshots are useful for moving files onto USB flash drives. You cannot use the **`copy`** command or any other file-moving technique to move files from an internal memory source to USB memory on the switch.
- You cannot use snapshots to move files to any destination outside of the switch other than an installed external USB flash drive or to move files between switches that are members of the same virtual chassis.
- The **`snapshot`** commands, similar to other virtual chassis commands, are always run on a local switch. In cases where different member switches of the same Virtual Chassis request the snapshot, the **`snapshot`** command is pushed to the Virtual Chassis member. This action creates the snapshot, executed, and the output is then returned to the switch that initiated the process.

  For example, the command to create an external snapshot on Virtual Chassis member 3 is entered from Virtual Chassis member 1. In this case, the snapshot of internal memory on Virtual Chassis member 3 is taken on external memory on Virtual Chassis member 3. The output of the process is seen from Virtual Chassis member 1. No files move between the switches.

Backing up IBM j-type m-series Ethernet routers

When the **`request system snapshot`** command is issued, the `/root` file system is backed up to the `/altroot` directory, and `/config` file system is backed up to the `/altconfig` directory. The `/root` and `/config` file systems are on the CompactFlash card of the router. The `/altroot` and `/altconfig` file systems are on the hard disk of the router. When the backup is completed, the current and backup software installations are identical.

On routers without a CompactFlash card, where the hard disk drive is the primary boot device, you cannot back up your software installation.

> **Important:** On systems with dual Routing Engines, perform this operation on both Routing Engines.

14.4.3 Upgrading the Junos software

The following sections explain the standard installation method, which is the typical method that is used to upgrade or downgrade software on the server. This method uses the installation package that matches the installation package that is already installed on the system. For any other types of upgrades, see *JUNOS Software Installation and Upgrade Guide,* GA32-0695.

Single Routing Engine

You can use the following procedure to upgrade the Junos software on an IBM j-type e-series Ethernet switch with a single Routing Engine. This method includes an individual member of a Virtual Chassis or all members of a Virtual Chassis, an IBM J08E or an IBM J16E using a single Routing Engine. The same procedure also applies to IBM j-type m-series Ethernet routers with a single Routing Engine.

1. Download the software package.
2. Optional: Back up the current software configuration as explained in 14.4.2, “System snapshot” on page 513.
3. Optional: Copy the software package to the switch or router. You can use FTP to copy the file to the `/var/tmp` directory.

   This step is optional because the Junos software can also be upgraded when the software image is stored at a remote location.
4. Install the new package on the device:

   ```
   user@switch> request system software add source reboot
   ```

   Replace *source* with one of the following paths:

   – For a software package that is installed from a local directory on the switch, use `/pathname/package-name-m.nZx-distribution.tgz`.

   – For a software package that is downloaded and installed from a remote location, use either of the following paths:

     - `ftp://hostname/pathname/package-name-m.nZx-distribution.tgz`
     - `http://hostname/pathname/package-name-m.nZx-distribution.tgz`

   A reboot, which occurs as part of the execution of the following command, is required to complete the software upgrade. If you want to reboot the switch at a later time, do not use the **reboot** option. Enter the **request system reboot** command at a later time to reboot the switch.

   Include the optional **member** option to install the software package on only one member of a Virtual Chassis. To install the software on all members of the Virtual Chassis, do not include the **member** option.
5. After the reboot is completed, log in and verify that the new version of the software is properly installed:

   ```
   user@switch> show version
   ```

Dual Routing Engine

For a switch or router with redundant Routing Engines, you can minimize disrupting network operation during a Junos software upgrade by upgrading the Routing Engines separately, starting with the backup Routing Engine.

Install the new Junos software release on the backup Routing Engine while keeping the currently running software version on the master Routing Engine. After verifying that the new software version is running correctly on the backup Routing Engine, switch device control to the backup Routing Engine. Finally, install the new software on the new backup Routing Engine.

Preparing the device for upgrade

Follow these steps to prepare the device for the upgrade:

1. Enter configuration mode:

   `user@switch> configure`

2. Disable Graceful Routing Engine Switchover (GRES):

   `user@switch# deactivate chassis redundancy graceful-switchover`

3. Save the configuration change on both Routing Engines:

   `user@switch# commit synchronize`

4. Exit the CLI configuration mode:

   `user@switch# exit`

5. Optional: Back up the current software configuration to a second storage option as described in 14.4.2, “System snapshot” on page 513.

Installing the software on the backup Routing Engine

Follow these steps to install the software on the backup Routing Engine:

1. Download the software package.
2. Copy the software package to the switch or router. You can use FTP to copy the file to the `/var/tmp` directory.
3. Log in to the console of the backup Routing Engine.
4. Install the new software package:

   `user@switch> request system software add validate /var/tmp/package-name-m.nZx-distribution.tgz`

   where *`package-name-m.nZx-distribution.tgz`* is the name of the package.

   > **Ending the installation:** To end the installation, do not restart your device. Instead, finish the installation and then issue the following command:
   >
   > `request system software delete package-name-m.nZx-distribution.tgz`
   >
   > where *`package-name-m.nZx-distribution.tgz`* is the name of the package. Running this command is your last chance to stop the installation.

5. Restart the new software:

   `user@switch> request system reboot`

6. After the reboot is completed, log in and verify that the new version of the software is properly installed:

   `user@switch> show version`

Installing the software on the default master Routing Engine

Follow the next steps to install the software on the default master Routing Engine:

1. Log in to the master Routing Engine console port.
2. Transfer device control to the backup Routing Engine:

   `user@switch> request chassis routing-engine master switch`

Important: Because GRES is disabled, this switchover causes all line cards in the switch to reload. All network traffic passing through these line cards is lost during the line card reloads.

3. Verify that the default backup Routing Engine is now the master Routing Engine:

```
user@switch> show chassis routing-engine

Routing Engine status:
  Slot 0:
    Current state Backup
    Election priority Master (default)
Routing Engine status:
  Slot 1:
    Current state Master
    Election priority Backup (default)
```

4. Install the new software package:

```
user@switch> request system software add validate
/var/tmp/jinstall-ex-8200-9.5R1.5-domestic-signed.tgz
```

5. Restart the Routing Engine:

```
user@switch> request system reboot
```

6. Wait for the prompt after the restart is completed.
7. Log in to the default backup Routing Engine (slot 1) through the console port.
8. Re-enable GRES:

```
user@switch# activate chassis redundancy graceful-switchover
```

By re-enabling GRES, any future Routing Engine switchovers can occur without the loss of any network traffic.

9. Save the configuration change:

```
user@switch# commit synchronize
```

10. Log in and verify the version of the software that is installed.

Returning routing control to the Routing Engine

If you want to return routing control to the Routing Engine that was the master Routing Engine at the beginning of the procedure, perform the next task:

1. Transfer routing control back to the default master Routing Engine:

```
user@switch> request chassis routing-engine master switch
```

2. Verify the change:

```
user@switch> show chassis routing-engine
```

Code examples of the software upgrade

The following examples show the completed software upgrade on IBM j-type m-series Ethernet router with dual Routing Engines. Example 14-16 shows the steps from “Preparing the device for upgrade” on page 516 on the default master Routing Engine.

Example 14-16 Preparation steps on the master Routing Engine

```
ibm@J11M-re0> show version
Hostname: J11M-re0
Model: mx960
JUNOS Base OS boot [9.1R1.8]
JUNOS Base OS Software Suite [9.1R1.8]
JUNOS Kernel Software Suite [9.1R1.8]
JUNOS Crypto Software Suite [9.1R1.8]
JUNOS Packet Forwarding Engine Support (M/T Common) [9.1R1.8]
JUNOS Packet Forwarding Engine Support (MX Common) [9.1R1.8]
JUNOS Online Documentation [9.1R1.8]
JUNOS Routing Software Suite [9.1R1.8]

{master}
ibm@J11M-re0> configure

{master}[edit]
ibm@J11M-re0# deactivate chassis redundancy graceful-switchover

{master}[edit]
ibm@J11M-re0# commit synchronize
re0:
configuration check succeeds
re1:
[edit groups re1 interfaces fxp0 unit 0 family inet]
  'address 10.1.1.42/24'
    warning: [edit system backup-router] not present. The default route for fxp0
is not installed.
[edit groups re1 interfaces fxp0 unit 0 family inet]
  'address 10.1.1.40/24'
    warning: [edit system backup-router] not present. The default route for fxp0
is not installed.
commit complete
ere0:
xcommit complete

[edit]
ibm@J11M-re0# exit
Exiting configuration mode

ibm@J11M-re0> request system snapshot
Verifying compatibility of destination media partitions...
Running newfs (899MB) on removable-compact-flash media / partition (ad2s1a)...
Running newfs (99MB) on removable-compact-flash media /config partition
(ad2s1e)...
Copying '/dev/ad0s1a' to '/dev/ad2s1a' .. (this may take a few minutes)
Copying '/dev/ad0s1e' to '/dev/ad2s1e' .. (this may take a few minutes)
The following filesystems were archived: / /config
```

Example 14-17 shows steps from “Installing the software on the backup Routing Engine” on page 516 performed on the default backup Routing Engine. For brevity, the **no-validate** option was used.

Example 14-17 Steps issued on the backup Routing Engine

```
ibm@J11M-re1> request system software add
/var/tmp/jinstall-9.5R4.3-domestic-signed.tgz no-validate
Installing package '/var/tmp/jinstall-9.5R4.3-domestic-signed.tgz' ...
Verified jinstall-9.5R4.3-domestic.tgz signed by PackageProduction_9_5_0
Adding jinstall...
Verified manifest signed by PackageProduction_9_5_0

WARNING:     This package will load JUNOS 9.5R4.3 software.
WARNING:     It will save JUNOS configuration files, and SSH keys
WARNING:     (if configured), but erase all other files and information
WARNING:     stored on this machine.  It will attempt to preserve dumps
WARNING:     and log files, but this can not be guaranteed.  This is the
WARNING:     pre-installation stage and all the software is loaded when
WARNING:     you reboot the system.

Saving the config files ...
NOTICE: uncommitted changes have been saved in
/var/db/config/juniper.conf.pre-install
Installing the bootstrap installer ...

WARNING:     A REBOOT IS REQUIRED TO LOAD THIS SOFTWARE CORRECTLY. Use the
WARNING:     'request system reboot' command when software installation is
WARNING:     complete. To abort the installation, do not reboot your system,
WARNING:     instead use the 'request system software delete jinstall'
WARNING:     command as soon as this operation completes.

Saving package file in /var/sw/pkg/jinstall-9.5R4.3-domestic-signed.tgz ...
Saving state for rollback ...

ibm@J11M-re1> request system reboot
```

Example 14-18 shows steps from “Installing the software on the default master Routing Engine” on page 516 that were performed on the default master Routing Engine. For brevity, the **no-validate** option was used.

Example 14-18 Step performed on the default master Routing Engine

```
root@J11M-re0> request chassis routing-engine master switch
warning: Traffic will be interrupted while the PFE is re-initialized
Toggle mastership between Routing Engines ? [yes,no] (no) yes

Resolving mastership...
Complete. The other Routing Engine becomes the master.

root@J11M-re0> show chassis routing-engine
Routing Engine status:
  Slot 0:
    Current state                  Backup
    Election priority              Master (default)
    Temperature                 41 degrees C / 105 degrees F
```

```
    CPU temperature             49 degrees C / 120 degrees F
    DRAM                      3584 MB
    Memory utilization          12 percent
    CPU utilization:
      User                       0 percent
      Background                 0 percent
      Kernel                     0 percent
      Interrupt                  0 percent
      Idle                     100 percent
    Model                          RE-S-2000
    Serial ID                      1000740797
    Start time                     2010-04-27 19:58:07 UTC
    Uptime                         8 days, 18 minutes, 8 seconds
    Load averages:                 1 minute   5 minute  15 minute
                                       0.12       0.06       0.01
Routing Engine status:
  Slot 1:
    Current state                  Master
    Election priority              Backup (default)
    Temperature                 43 degrees C / 109 degrees F
    CPU temperature             50 degrees C / 122 degrees F
    DRAM                      3584 MB
    Memory utilization          11 percent
    CPU utilization:
      User                       0 percent
      Background                 0 percent
      Kernel                     3 percent
      Interrupt                  0 percent
      Idle                      96 percent
    Model                          RE-S-2000
    Serial ID                      1000740866
    Start time                     2010-05-05 20:13:04 UTC
    Uptime                         3 minutes, 15 seconds

root@J11M-re0> request system software add
/var/tmp/jinstall-9.5R4.3-domestic-signed.tgz no-validate
Installing package '/var/tmp/jinstall-9.5R4.3-domestic-signed.tgz' ...
Verified jinstall-9.5R4.3-domestic.tgz signed by PackageProduction_9_5_0
Adding jinstall...
Verified manifest signed by PackageProduction_9_5_0

WARNING:     This package will load JUNOS 9.5R4.3 software.
WARNING:     It will save JUNOS configuration files, and SSH keys
WARNING:     (if configured), but erase all other files and information
WARNING:     stored on this machine.  It will attempt to preserve dumps
WARNING:     and log files, but this can not be guaranteed.  This is the
WARNING:     pre-installation stage and all the software is loaded when
WARNING:     you reboot the system.

Saving the config files ...
NOTICE: uncommitted changes have been saved in
/var/db/config/juniper.conf.pre-install
Installing the bootstrap installer ...

WARNING:     A REBOOT IS REQUIRED TO LOAD THIS SOFTWARE CORRECTLY. Use the
```

```
WARNING:     'request system reboot' command when software installation is
WARNING:     complete. To abort the installation, do not reboot your system,
WARNING:     instead use the 'request system software delete jinstall'
WARNING:     command as soon as this operation completes.

Saving package file in /var/sw/pkg/jinstall-9.5R4.3-domestic-signed.tgz ...
Saving state for rollback ...

root@J11M-re0> request system reboot
Reboot the system ? [yes,no] (no) yes

Shutdown NOW!
```

For more information about upgrading Junos software on the IBM j-type m-series Ethernet router, see *Installation and Upgrade Guide*, GA32-0695.

For more information about upgrading Junos software on the IBM j-type e-series Ethernet switch, see *Complete Software Guide for JUNOS Software for EX Series Ethernet Switches, Release 10.1*.

14.5 Overview of the file system

The following directories are the most important ones in the Junos software file system:

/ The root file system on the boot device (usually the primary compact-flash drive).

/config Located on the boot device; contains the current operational configuration, the last three committed configurations, and the rescue configuration.

/config/db/config Contains up to 46 additional previous committed configurations.

/var Contains the following subdirectories.

- `/var/home` Contains users home directories, created when you create user accounts.
- `/var/log` Contains system logs and tracing files.
- `/var/tmp` Contains daemon core files and temporary files.

> **Important:** Junos software deletes the contents of the `/var` directory during software upgrades. Therefore, you must back up the required files and logs externally.

You can view the directory structure of the device and the individual files by issuing the **`file`** command in operational mode as shown in Example 14-19 on page 522. Use the **`list`** option to see the directory structure of the router.

The default directory for the **`file list`** command is the home directory of the user logged in to the router. In fact, the home directory of the user is the default directory for most of Junos software commands that require a file name.

Example 14-19 The file command

```
ibm@J11M-re0> file ?
Possible completions:
  <[Enter]>            Execute this command
  archive              Archives files from the system
  checksum             Calculate file checksum
  compare              Compare files
  copy                 Copy files (local or remote)
  delete               Delete files from the system
  list                 List file information
  rename               Rename files
  show                 Show file contents
  source-address       Local address to use in originating the connection
  |                    Pipe through
{master}
ibm@J11M-re0> file list

/var/home/ibm/:
.ssh/
```

To view the contents of other file directories, specify the directory location, as shown in Example 14-20.

Example 14-20 The file list command

```
ibm@J11M-re0> file list /config

/config:
.snap/
juniper.conf.1.gz
juniper.conf.2.gz
juniper.conf.3.gz
juniper.conf.gz
license/
license.old/
rescue.conf.gz
usage.db
```

Example 14-21 shows how to use the context-sensitive help system of the router to locate a directory.

Example 14-21 Locating a directory

```
ibm@J11M-re0> file list /?
Possible completions:
  <[Enter]>            Execute this command
  <path>               Path to list
  /COPYRIGHT           Size: 6187, Last changed: Apr 01 2007
  /altconfig/          Last changed: Apr 25 2008
  /altroot/            Last changed: Apr 25 2008
  /bin/                Last changed: May 19 2008
  /boot/               Last changed: Apr 28 18:57:14
  /config/             Last changed: Apr 29 16:18:55
  /data/               Last changed: Apr 25 2008
  /dev/                Last changed: Apr 27 19:58:10
```

```
  /etc/                Last changed: Apr 28 18:57:14
  /kernel              Size: 54440445, Last changed: Apr 25 2008
  /mfs/                Last changed: Apr 27 19:58:23
  /mnt/                Last changed: Jan 12 2007
  /modules/            Last changed: May 19 2008
  /opt/                Last changed: May 19 2008
  /packages/           Last changed: May 19 2008
  /proc/               Last changed: May 03 19:55:23
  /root/               Last changed: May 19 2008
  /sbin/               Last changed: May 19 2008
  /staging/            Last changed: Apr 27 20:04:36
  /tmp/                Last changed: Apr 28 18:57:14
  /usr/                Last changed: May 19 2008
  /var/                Last changed: Apr 27 19:58:23
```

To view the contents of a file, issue the **file show** command (Example 14-22).

Example 14-22 The file show command

```
ibm@J11M-re0> file show /var/log/inventory
Apr  2 02:29:07 CHASSISD release 9.0R1.10 built by builder on 2008-02-14 03:21:01
UTC
Apr  2 02:29:07 Midplane - part number 710-017414, serial number TR3118
Apr  2 02:29:07 CB - part number 710-021523, serial number KE0371
Apr  2 02:29:08 mx480 chassis, serial number JN110059BAFB
```

14.6 Managing the configuration

This section describes the most important tasks to manage the configuration in Junos software.

14.6.1 Saving the configuration

When you edit a configuration, you work in a copy of the current configuration to create a candidate configuration. The changes that you make to the candidate configuration are visible in the CLI immediately. Therefore, if multiple users are editing the configuration at the same time, all users can see all changes.

To have a candidate configuration take effect, you commit the changes. The candidate file is checked for the correct syntax, activated, and marked as the current, operational software configuration file. If multiple users are editing the configuration, when you commit the candidate configuration, all changes made by all the users take effect.

In addition to saving the current configuration, the CLI saves the current operational version and the previous 49 versions of committed configurations. The most recently committed configuration is version 0. This version is the current operational version, which is the default configuration that the system returns to if you roll back to a previous configuration. The oldest saved configuration is version 49.

The currently operational Junos software that is configuration is stored in the `juniper.conf` file, and the last three committed configurations are stored in the `juniper.conf.1`, `juniper.conf.2`, and `juniper.conf.3` files. These four files are in the `/config` directory, which is on the flash drive of the router. The remaining 46 previous versions of committed

configurations, the `juniper.conf.4` through `juniper.conf.49` files, are stored in the directory `/var/db/config` directory on the hard disk.

14.6.2 Returning to the most recently committed configuration

To return to the most recently committed configuration and load it into configuration mode without activating it, use the **rollback** configuration mode command:

```
ibm@J11M-re0# rollback
```

To activate the configuration to which you rolled back, use the **commit** command:

```
ibm@J11M-re0# commit
```

14.6.3 Returning to a previously committed configuration

To return to a configuration before the most recently committed one, include the number in the **rollback** command. The most recently saved configuration is 0, which is the default configuration to which the system returns. The oldest saved configuration is number 49.

```
ibm@J11M-re0# rollback 3
```

14.6.4 Viewing previous configurations

To view previous configurations, including the rollback number, date, time, the name of the user who committed changes, and the method of commit, use the **rollback ?** command as shown in Example 14-23.

Example 14-23 Rollback options

```
ibm@J11M-re0# rollback ?
Possible completions:
  <[Enter]>            Execute this command
  0                    2010-04-28 18:57:14 UTC by ibm via cli commit synchronize
  1                    2010-04-27 20:04:36 UTC by root via cli commit synchronize
  2                    2010-04-27 20:04:11 UTC by root via cli commit synchronize
  3                    2010-04-27 20:03:23 UTC by root via cli commit synchronize
  4                    2010-04-27 20:00:58 UTC by root via cli commit synchronize
  5                    2010-04-27 20:00:38 UTC by root via cli
  6                    2010-04-27 19:56:57 UTC by root via cli
  7                    2010-04-27 19:50:32 UTC by root via cli
  8                    2009-08-07 14:05:02 UTC by nickv via synchronize
  9                    2009-07-29 12:20:09 UTC by root via other
  10                   2009-07-15 14:42:37 UTC by nickv via synchronize
  11                   2009-07-10 19:20:37 UTC by nickv via synchronize
  12                   2009-07-10 19:19:07 UTC by nickv via synchronize
  13                   2009-07-10 19:14:28 UTC by nickv via synchronize
  14                   2009-07-10 16:04:16 UTC by nickv via synchronize
  15                   2009-07-10 16:02:44 UTC by nickv via synchronize
  16                   2009-07-10 16:01:43 UTC by nickv via synchronize
  17                   2009-07-10 15:55:44 UTC by nickv via synchronize
  18                   2009-06-15 19:28:28 UTC by root via other
  19                   2009-04-29 16:24:28 UTC by sstrattn via synchronize
  20                   2009-01-19 15:19:13 UTC by sstrattn via synchronize
  21                   2009-01-19 15:18:17 UTC by sstrattn via synchronize
  22                   2009-01-19 14:30:51 UTC by sstrattn via synchronize
```

```
  23                    2008-06-24 14:27:22 UTC by sstrattn via synchronize
  24                    2008-06-13 17:39:08 UTC by nickv via synchronize
  25                    2008-06-13 15:07:57 UTC by nickv via synchronize
  26                    2008-06-12 20:21:20 UTC by nickv via synchronize
  27                    2008-06-12 20:15:35 UTC by nickv via synchronize
  28                    2008-06-12 20:13:25 UTC by nickv via synchronize
  29                    2008-06-11 06:11:03 UTC by root via other
  30                    2008-06-09 19:02:25 UTC by root via cli commit synchronize
  31                    2008-05-19 19:06:12 UTC by lab via cli commit synchronize
  32                    2008-05-19 19:02:41 UTC by lab via cli commit synchronize
  33                    2008-05-19 18:59:02 UTC by lab via cli commit synchronize
  34                    2008-05-19 18:55:57 UTC by lab via cli commit synchronize
  35                    2008-05-19 15:01:53 UTC by root via cli commit synchronize
  36                    2008-05-19 15:00:11 UTC by root via cli commit synchronize
  37                    2008-05-19 14:56:10 UTC by root via cli commit synchronize
  38                    2008-05-19 14:54:21 UTC by root via cli commit synchronize
  39                    2008-05-19 14:52:52 UTC by root via synchronize
  40                    2008-05-19 14:50:17 UTC by root via other
  41                    2008-05-19 14:50:14 UTC by root via cli commit synchronize
  42                    2008-05-19 14:48:35 UTC by root via other
  43                    2008-05-19 14:48:14 UTC by root via other
  44                    2008-05-19 14:38:35 UTC by juniper via synchronize
  45                    2008-05-19 14:14:00 UTC by juniper via synchronize
  46                    2008-05-19 13:51:11 UTC by juniper via synchronize
  47                    2008-05-19 13:29:25 UTC by juniper via synchronize
  48                    2008-05-19 13:27:05 UTC by juniper via cli commit
synchronize
  49                    2008-05-19 12:15:37 UTC by root via synchronize
  rescue                2010-04-29 16:18:55 UTC by ibm via cli
  |                     Pipe through a command
```

14.6.5 Comparing configuration changes

In configuration mode only, after you change the configuration and want to compare the candidate configuration with a prior version, you can use the **compare** command to view the configuration. The **compare** command compares the candidate configuration with either the current committed configuration or a configuration file. This command also shows the differences between the two configurations. To compare configurations, specify the **compare** command after the pipe:

`ibm@J11M-re0#` **show** | **compare** (*filename* | **rollback** *n*)

where:

filename is the full path to a configuration file. The file must be in the correct format, which is a hierarchy of statements.

n is the index into the list of previously committed configurations. The most recently saved configuration is 0, and the oldest saved configuration is number 49. If you do not specify arguments, the candidate configuration is compared against the active configuration file (`/config/juniper.conf`).

The comparison output uses the following conventions:

- Statements that are only in the candidate configuration are prefixed with a plus sign (+).
- Statements that are only in the comparison file are prefixed with a minus sign (–).
- Statements that are unchanged are prefixed with a single blank space ( ).

Example 14-24 shows the comparison between the active configuration and rollback 3 configuration.

Example 14-24 Configuration comparison

```
ibm@J11M-re0# show | compare rollback 3
[edit groups re1 interfaces]
-    me0 {
-        unit 0 {
-            family inet;
-        }
-    }
[edit groups re0 interfaces]
-    me0 {
-        unit 0 {
-            family inet;
-        }
-    }
[edit system services]
+    ssh {
+        protocol-version v2;
+    }
```

14.6.6 Saving a configuration to a file

You might want to save the configuration to a file so that you can edit it with a text editor of your choice. You can save your current configuration to an ASCII file, which saves the configuration in its current form, including any uncommitted changes. If more than one user is modifying the configuration, all changes made by all users are saved.

To save software configuration changes to an ASCII file, use the **save** configuration mode command:

```
ibm@J11M-re0# save day1config
Wrote 77 lines of configuration to 'day1config'
```

The contents of the current level of the statement hierarchy (and below) are saved, along with the statement hierarchy that contains it. By using this method, a section of the configuration can be saved, while fully specifying the statement hierarchy. By default, the configuration is saved to a file in your home directory.

When you issue the **save** command from anywhere in the hierarchy (except the top level), a *replace* tag is automatically included at the beginning of the file. You can use the *replace* tag to control how a configuration is loaded from a file.

14.6.7 Loading a configuration from a file

You can create a file, copy the file to the local router, and then load the file into the CLI. After you load the file, you can commit it to activate the configuration on the router, or you can edit the configuration interactively using the CLI and commit it at a later time.

You can also create a configuration, while typing at the terminal, and then load it. Loading a configuration from the terminal is generally useful when you are cutting existing portions of the configuration and pasting them elsewhere in the configuration.

To load an existing configuration file that is on the router, use the **load** configuration mode command. Example 14-25 shows the different options that are available with this command.

Example 14-25 The load command

```
ibm@J11M-re0# load ?
Possible completions:
  factory-default      Override existing configuration with factory default
  merge                Merge contents with existing configuration
  override             Override existing configuration
  patch                Load patch file into configuration
  replace              Replace configuration data
  set                  Execute set of commands on existing configuration
  update               Update existing configuration

ibm@J11M-re0# load override ?
Possible completions:
  <filename>           Filename (URL, local, remote, or floppy)
  terminal             Use login terminal
```

To load a configuration from the terminal, specify the **terminal** option. Press Ctrl+D to end the input.

To replace an entire configuration, specify the **override** option at any level of the hierarchy. The **override** option discards the current candidate configuration and loads the configuration in *filename* or the one that you type at the terminal. When you use the **override** option and commit the configuration, all system processes reparse the configuration.

To replace portions of a configuration, specify the **replace** option. For this operation to work, you must include *replace:* tags in the file or configuration that you type at the terminal. The software searches for the *replace:* tags, deletes the existing statements of the same name, if any, and replaces them with the incoming configuration. If no existing statement of the same name exists, the **replace** operation adds to the configuration the statements marked with the *replace:* tag.

To replace only the configuration that has changed, specify the **update** option at any level of the hierarchy. The **update** option compares the current configuration and the current candidate configuration. It then loads only the changes between these configurations in *filename* or the one that you type at the terminal. When you use the **update** option and commit the configuration, the Junos software attempts to notify the smallest set of system processes that are affected by the configuration change.

To combine the current configuration and the configuration in *filename* or the one that you type at the terminal, specify the **merge** option. The **merge** option is useful when you are adding a new section to an existing configuration. If the existing configuration and the incoming configuration contain conflicting statements, the statements in the incoming configuration override the statements in the existing configuration.

To change part of the configuration with a patch file and mark only those parts as changed, specify the **patch** option.

For more information about this topic, see *JUNOS Software CLI User Guide,* GA32-0697-02.

14.7 Chassis and interfaces alarms

This section provides information about the different classes of chassis and interfaces alarms for the IBM j-type m-series Ethernet router family. For information about this topic on IBM j-type e-series Ethernet switches, see Chapter 13, “System management” on page 409.

When the Routing Engine detects an alarm condition, it lights the red or yellow alarm LED on the craft interface as appropriate. To view a more detailed description of the alarm cause, use the **show chassis alarms** command (Example 14-26).

Example 14-26 The show chassis alarms command

```
{master}
ibm@J11M-re0> show chassis alarms
5 alarms currently active
Alarm time               Class  Description
2010-05-05 20:31:20 UTC  Major  Too Few AC PEMs
2010-05-05 20:31:06 UTC  Major  PEM 1 Input Failure
2010-05-05 20:31:06 UTC  Major  PEM 1 Not OK
2010-05-05 20:31:06 UTC  Major  PEM 0 Input Failure
2010-05-05 20:31:06 UTC  Major  PEM 0 Not OK
```

The **show chassis alarms** command has two classes of alarm messages:

- *Chassis alarms* indicate a problem with a chassis component such as the cooling system or power supplies.
- *Interface alarms* indicate a problem with a specific network interface.

14.7.1 Alarm relay contacts

The craft interface has two alarm relay contacts for connecting the router to external alarm devices. Whenever a system condition triggers either the red or yellow alarm on the craft interface, the alarm relay contacts are also activated. The alarm relay contacts are on the upper right side of the craft interface.

14.7.2 Craft interface LEDs

The craft interface is the panel on the front of the router above the dense port concentrator cards that contain LEDs and buttons so that you can troubleshoot the router.

The craft interface includes the following LEDs:

Alarm LEDs
One large red circular LED and one large yellow triangular LED, on the upper right corner of the craft interface, indicate two levels of alarm conditions. The circular red LED lights to indicate a critical condition can result in a system shutdown. The triangular yellow LED lights to indicate a less severe condition requires monitoring or maintenance. Both LEDs can be lit simultaneously. A condition that causes an alarm LED to light also activates the corresponding alarm relay contact on the craft interface.

Host subsystem LEDs
Three LEDs, MASTER, ONLINE, and OFFLINE, indicate the status of the host subsystem. A green MASTER LED indicates that the host is functioning as the master. The ONLINE LED indicates that the host is

online. The OFFLINE LED indicates that the host is installed but the Routing Engine is offline. The host subsystem LEDs are on the left side of the craft interface and are labeled RE0 and RE1.

Power supply LEDs Two LEDs (PEM) indicate the status of each power supply. Green indicates that the power supply is functioning normally. Red indicates that the power supply is not functioning normally. The power supply LEDs are in the center craft interface and are labeled 0 through 3.

Line card LEDs Two LEDs, OK and FAIL, indicate the status of each dense port concentrator, FPC, or MPC. Green indicates OK, and red indicates a failure. The line card LEDs are located along the bottom of the craft interface.

Switch control board LEDs
Two LEDs, OK and FAIL, indicate the status of each switch control board. Green indicates OK, and red indicates a failure. The switch control board LEDs are on the left side of the craft interface along the bottom.

Fan LEDs Two LEDs indicate the status of the fans. Green indicates that the fans are functioning normally, and red indicates that a fan has failed. The fan LEDs are on the upper left side of the craft interface.

14.7.3 Component LEDs

The following LEDs are on various router components and show the status of those components:

Dense port concentrator LED
One LED labeled OK/FAIL on each dense port concentrator faceplate indicates the status of the dense port concentrator.

FPC LED One LED labeled OK/FAIL on each FPC faceplate indicates the FPC's status.

MPC LED One LED labeled OK/FAIL on each FPC faceplate indicates the FPC's status.

MIC LED One LED labeled OK/FAIL on each MIC faceplate indicates the MIC's status.

PIC LED One LED labeled OK/FAIL on each PIC faceplate indicates the PIC's status.

Switch control board LEDs
Three LEDs, labeled FABRIC ACTIVE, FABRIC ONLY, and OK/FAIL, on each switch control board faceplate indicate the status of the switch control board. If no LEDs are lit, the master Routing Engine might still be starting or the switch control board is not receiving power.

Routing Engine LEDs
Four LEDs, labeled MASTER, HDD, ONLINE, and FAIL, on each Routing Engine faceplate indicate the status of the Routing Engine and hard disk drive.

Power supply LEDs Two LEDs on each power supply faceplate indicate the status of that power supply.

14.8 More information

For more information, see the following guides:

- For Junos software CLI, see *JUNOS Software CLI User Guide,* GA32-0697-02:

  http://www-01.ibm.com/support/docview.wss?rs=1331&context=SGPMK5&dc=DA400&uid=isg3T7000171&loc=en_US&cs=utf-8&lang=en

- For upgrading Junos software on IBM j-type m-series, see *Installation and Upgrade Guide,* GA32-0695:

  http://www-01.ibm.com/support/docview.wss?rs=1335&context=SGPMHJ&dc=DA400&uid=isg3T7000150&loc=en_US&cs=utf-8&lang=en

- For Junos software on IBM j-type e-series, see *Complete Software Guide for JUNOS Software for EX Series Ethernet Switches, Release 10.1*, at the following web page:

  http://www.juniper.net/techpubs/en_US/junos10.1/information-products/topic-collections/ex-series/software-all/book-software-ex-series-101-all.pdf

- For hardware maintenance and troubleshooting for IBM Ethernet Router J02M, *IBM Ethernet Router J02M Hardware Guide,* GA32-0655:

  http://www-01.ibm.com/support/docview.wss?rs=1331&context=SGPMK5&dc=DA400&uid=isg3T7000128&loc=en_US&cs=utf-8&lang=en

- For hardware maintenance and troubleshooting for IBM Ethernet Router J06M, see *IBM Ethernet Router J06M Hardware Guide,* GA32-0657:

  http://www-01.ibm.com/support/docview.wss?rs=1333&context=SGPMLA&dc=DA400&uid=isg3T7000130&loc=en_US&cs=utf-8&lang=en

- For hardware maintenance and troubleshooting for IBM Ethernet Router J11M, see *IBM Ethernet Router J11M Hardware Guide,* GA32-0659:

  http://www-01.ibm.com/support/docview.wss?rs=1334&context=SGPMM5&dc=DA400&uid=isg3T7000133&loc=en_US&cs=utf-8&lang=en

- For hardware maintenance and troubleshooting for IBM Ethernet Switch J08E, see *IBM Ethernet Switch J08E Complete Hardware Guide,* GA32-0665-03:

  http://www-01.ibm.com/support/docview.wss?rs=1335&context=SGPMHJ&dc=DA400&uid=isg3T7000122&loc=en_US&cs=utf-8&lang=en

- For hardware maintenance and troubleshooting for IBM Ethernet Switch J16E, see *IBM Ethernet Switch J16E Hardware Guide,* GA32-0667-03:

  http://www-01.ibm.com/support/docview.wss?rs=1336&context=SGPMJD&dc=DA400&uid=isg3T7000124&loc=en_US&cs=utf-8&lang=en

- For hardware maintenance and troubleshooting for IBM Ethernet Switch J48E, see *IBM Ethernet Switch J48E Hardware Guide,* GA32-0663-04:

  http://www-01.ibm.com/support/docview.wss?rs=1337&context=SGPMMT&dc=DA400&uid=isg3T7000126&loc=en_US&cs=utf-8&lang=en

A

Preferred practices for networking

This appendix describes the preferred practices that can help improve the overall networking reliability, security, and performance. They are based on experience. The evaluation of alternative practices has demonstrated repeatable success and can be replicated in most environments. These practices might apply in every network environment. When using a practice that contradicts a preferred practice, carefully evaluate the affect of the practice on the entire networking environment before implementing it.

This appendix includes the following topics:

- Monitoring and management
- Traffic isolation and reduction
- Cabling
- Port configuration
- Additional resources

© Copyright IBM Corp. 2011. All rights reserved.

Monitoring and management

For monitoring and management of your network, consider the following practices:

- To allow the router to start and to ensure that the router is reachable over the network, configure a backup router, which is a router that is on the same local subnet. This practice is preferred because, while the router is starting, the routing protocol process is not running, and the router has no static or default routes.

  When the routing protocols start, the address of the backup router is removed from the local routing and forwarding tables. To have the address remain in these tables, configure a static route.
- Separate management traffic from production traffic. You can accomplish this task by using out-of-band management in which you use a separate network for the management traffic. The alternative is to use inband management in which you use the same links as production traffic. When inband management is implemented, use a separate management virtual local area network (VLAN) to logically isolate the management traffic from production traffic.
- Use a log (syslog) server to record significant events that occur in the network. System event logs are the primary source of information about such events. Determining which events must be logged depends on many factors and varies for each network. Generally, these events can be divided into the following categories:
  - Authentication and command events

    Recording when authentication and authorization is granted and rejected, and all user commands, provides excellent tracking of all management activity on the switches and routers.
  - Routing protocol events and errors

    Recording protocol events and errors can indicate attacks against routing protocols. Logs can show authentication failures that might point to an attacker sending spoofed or malformed routing packets.
  - Denial of traffic

    Recording all traffic that is denied access to the router can indicate other types of attacks in the network and the type and source of the attack.
- Use a Network Time Protocol (NTP) server and configured devices to use NTP to synchronize time on devices. Debugging and troubleshooting are much easier when the time stamps in the log files of all switches and routers are synchronized. When possible use NTP with authentication enabled to prevent unauthorized attempts to change time.
- Use firewall filters to restrict and limit the rate of traffic to the management interfaces of the switches and routers. Limit the traffic from only stations that are approved for management of the network. Also limit the rate of traffic to prevent denial of service (DoS) attacks.
- Use Secure Shell (SSH) and Secure Copy Protocol (SCP) for remote access and file copy to network switches and routers. Use of secure protocols for management are especially important when communication spans an untrusted network.

Traffic isolation and reduction

For traffic isolation and reduction, consider the following practices:

- Use VLANs within the network to limit broadcast domains. The extent of VLAN use varies with each network.
- Only configure links with the required VLANs. Having additional VLANs configured on links to switches that do not have clients or adjacent switches with ports configured on that VLAN allow unnecessary traffic to consume bandwidth resources on the link.
- Disable any protocols that are not being used on devices, such as network attached printers and servers. Protocols, such as AppleTalk, IPX, and NetBIOS, are commonly enabled by default on network attached printers or print server devices. Disabling or removing these unused protocols eliminates their broadcast affect in the network and the device attached.
- Use a separate VLAN for traffic using jumbo frames and configure the VLAN on separate ports to avoid any impact to sensitive traffic. The usage of jumbo frames on the same network ports as traffic for latency sensitive applications can affect performance of these applications.

Cabling

For cabling, consider the following practices:

- Disable ports or disconnect cables before routers or switches are properly configured for adding to the network. Connecting devices in their default configuration to a network can cause undesirable results. Properly configure the device, and then enable or connect the cables.
- Do not mix 50 micron and 62.5 micron fiber optic cabling. Mixing fiber optic cables with different core diameters in the same segment results in a loss of signal quality.
- Label each end of the cable with the source and destination.
- Use caution if securing cables with zip ties. Zip ties can exert excessive pressure on the cable, which especially applies to fiber optic cabling.
- Do not use cables longer than necessary. Excessive cable lengths make proper cable management difficult.
- Group cables in small bundles (4-6) if possible. Apply wraps along the bundle to keep the bundles together. Usually 18–24 in. intervals between wraps are sufficient.
- Use cabling trays and cable guides provided by racks or equipment where possible.
- Keep cable connectors covered and ports plugged when they are not installed to prevent dust contamination that might affect signal quality.
- Do not exceed the bend radius for fiber cabling, which can result in a loss of signal quality. The bend radius is determined by the cable manufacturer. However, generally the bend radius is 10 times the diameter of the cables. A cable with a diameter of 3 mm might be near 30 mm.
- Be careful that rack doors or other moveable devices do not apply excessive force or bind cables.
- Route cables in a manner that supports the weight of the cables without causing cable strain. Allow sufficient slack in the cabling to avoid strain. Use extra care in areas of high temperatures that cause cables to be flexible.

- Avoid routing cables over 90-degree angles. If cables are loose, this type of routing is not an issue. However, if the cables shift or are tightened, then the tight bend might damage the cabling.
- Avoid running copper Ethernet cables within 12 inches of power cables to eliminate the risk of electromagnetic interference (EMI) that can affect Ethernet signal quality. If you maintaining a sufficient distance from power cabling is not possible, consider using shielded or fiber optic cabling.

Port configuration

For port configuration, consider the following practices:

- When configuring 10/100 Mb links, configure ports on the device and switch port to the same settings. If you are using autonegotiation, you must set the device network interface card (NIC) and the switch port to *autonegotiation*. If static speed and duplex settings are used, configure both the device NIC and the switch port for the desire speed and duplex setting.

  When you do not follow this practice, a condition known as *duplex mismatch* often results (Table A-1).

  Table A-1 Duplex mismatch

Switch duplex		Device duplex	
Configured	**Observed**	**Configured**	**Observed**
Auto	Full	Auto	Full
Full	Full	Full	Full
Half	Half	Half	Half
Auto	Half	Half	Half
Auto	Half	Full	Full
Half	Half	Auto	Half
Full	Full	Auto	Half

  When a duplex mismatch condition occurs, traffic continues to be passed, but poor performance is noticed. The interface of the link operating in half duplex mode indicates collisions and excessive collisions. The interface of the link operating in full duplex mode indicates Cyclic Redundancy Check (CRC) errors.

- Use features that help prevent rogue devices from impacting the spanning tree within a network. Such features as Bridge Protocol Data Units (BPDU) guard and root guard are excellent tools to help protect the network from this type of threat.

Additional resources

See the following resources for additional information:

- Network Reliability and Interoperability Council (NRIC) Best Practices

  https://www.fcc.gov/nors/outage/bestpractice/BestPractice.cfm
- *Campus LAN Design Guide*

  http://www.juniper.net/us/en/local/pdf/design-guides/8020001-en.pdf
- *Virtual Chassis Technology Best Practices*

  http://kb.juniper.net/index?page=content&id=TN31
- *Implementing L2 at the Data Center Access Layer on Juniper Networks Infrastructure*

  http://www.juniper.net/us/en/local/pdf/implementation-guides/8010014-en.pdf
- *Implementing L3 at the Data Center Access Layer on Juniper Networks Infrastructure*

  http://www.juniper.net/us/en/local/pdf/implementation-guides/8010022-en.pdf
- *Deploying Fixed-Configuration and Chassis-Based EX Series Ethernet Switches in Campus LANs*

  http://www.juniper.net/us/en/local/pdf/implementation-guides/8010021-en.pdf
- *Securing Flows Within the Cloud-Ready Data Center Implementation Guide*

  http://www.juniper.net/us/en/local/pdf/implementation-guides/8010034-en.pdf
- *Implementing IBM POWERVM Virtual Machines on Juniper Networks Data Center Networks*

  http://www.juniper.net/us/en/local/pdf/implementation-guides/8010049-en.pdf

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

IBM Redbooks

The following IBM Redbooks publications provide additional information about the topic in this document. Note that some publications referenced in this list might be available in softcopy only.

- *IBM j-type Data Center Networking Introduction*, SG24-7820
- *IBM j-type Ethernet Appliance Implementation*, SG24-7883

You can search for, view, or download Redbooks, Redpapers, Web Docs, draft, and additional materials, as well as order hardcopy Redbooks publications, at the following website:

ibm.com/redbooks

Other publications

These publications are also relevant as further information sources:

- IBM j-type Ethernet switches and related hardware documentation
  - *IBM Ethernet Switch J48E Complete Hardware Guide,* GA32-0663
  - *IBM Ethernet Switch J48E Quick Start,* GA32-0664
  - *IBM Ethernet Switch J08E Complete Hardware Guide,* GA32-0665
  - *IBM Ethernet Switch J08E Quick Start,* GA32-0666
  - *IBM Ethernet Switch J16E Complete Hardware Guide,* GA32-0667
  - *IBM Ethernet Switch J16E Quick Start,* GA32-0668
- IBM j-type Ethernet routers and related hardware documentation
  - *IBM j-type m-series Ethernet Router Dense Port Concentrators (DPC) Guide,* GA32-0661
  - *IBM j-type m-series Ethernet Routing Engine Installation Instructions,* GA32-0682
  - *IBM j-type m-series Ethernet Services PIC Guide,* GA32-0662
  - *IBM Ethernet Router J02M Hardware Guide,* GA32-0655
  - *IBM Ethernet Router J02M Quick Start,* GA32-0656
  - *IBM Ethernet Router J06M Hardware Guide,* GA32-0657
  - *IBM Ethernet Router J06M Quick Start,* GA32-0658
  - *IBM Ethernet Router J11M Hardware Guide,* GA32-0659
  - *IBM Ethernet Router J11M Quick Start,* GA32-0660
  - *JUNOS Software IBM j-type m-series Ethernet Routers Solutions Guide,* GA32-0683

© Copyright IBM Corp. 2011. All rights reserved.

- Junos software documentation
  - *Juniper Web Device Manager for IBM j-type Ethernet Switches and Routers Interface User Guide,* GA32-0688
  - *JUNOS Software Access Privilege Configuration Guide,* GA32-0696
  - *JUNOS Software Broadband Subscriber Management Solutions Guide,* GA32-0709
  - *JUNOS Software Class of Service Configuration Guide,* GA32-0738
  - *JUNOS Software CLI User Guide,* GA32-0697
  - *JUNOS Software Configuration and Diagnostic Automation Guide,* GA32-0679
  - *JUNOS Software Ethernet Routing Engine Media Upgrade Kit,* GA32-0681
  - *JUNOS Software Feature Guide,* GA32-0739
  - *JUNOS Software Hierarchy and RFC Reference,* GA32-0712
  - *JUNOS Software High Availability Configuration Guide,* GA32-0670
  - *JUNOS Software IBM j-type m-series Ethernet Routers Layer 2 Configuration Guide,* GA32-0708
  - *JUNOS Software Installation and Upgrade Guide,* GA32-0695
  - *JUNOS Software Interfaces Command Reference,* GA32-0672
  - *JUNOS Software JUNOScript API Guide,* GA32-0674
  - *JUNOS Software MPLS Applications Configuration Guide,* GA32-0702
  - *JUNOS Software Multicast Protocols Configuration Guide,* GA32-0703
  - *JUNOS Software NETCONF API Guide,* GA32-0678
  - *JUNOS Software Network Interfaces Configuration Guide,* GA32-0706
  - *JUNOS Software Network Management Configuration Guide,* GA32-0698
  - *JUNOS Software Policy Framework Configuration Guide,* GA32-0704
  - *JUNOS Software Routing Protocols and Policies Command Reference,* GA32-0673
  - *JUNOS Software Services Interfaces Configuration Guide,* GA32-0707
  - *JUNOS Software Subscriber Access Configuration Guide,* GA32-0711
  - *JUNOS Software System Basics and Services Command Reference,* GA32-0671
  - *JUNOS Software System Log Messages Reference,* GA32-0675
  - *JUNOS Software VPNs Configuration Guide,* GA32-0705
  - *JUNOScope Software User Guide,* GA32-0670
  - *JUNOS Software Policy Framework Configuration Guide*, GA32-0704

Online resources

These websites are also relevant as further information sources:

- IBM Juniper Partner website

  https://simplifymydatacenter.com/ibm
- Product page for IBM j-type Ethernet switches and routers

  http://www.ibm.com/systems/networking/hardware/j-type/index.html
- Juniper Networks Support site

  http://www.juniper.net/customers/support/

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

Numerics

1024-bit RSA private key 145
4273-E48 41
4274-E08 41
4274-E16 41
4274-M02 51
4274-M06 51
4274-M11 51
802.1Q VLAN tag 238
802.1X
  authentication 263, 390, 404
  features 392
  standard 390
  supplicant 400

A

ac power circuit breaker 74
ac power cord 69, 77, 80
  specifications 73
ac power supply 69–70, 73, 75, 77, 79
  LEDs 69, 76, 80
access layer 64
access port 239, 381
access privileges 434
access security 367–368
accounting management 410
active configuration 111, 115, 160
active files 167
active path 303
active route 303
adaptive services (AS)
  boundary routers 289
  confederation 312
  path 301
adding a string 474
address mask 443
Address Resolution Protocol (ARP) 33
  cache 33
  entries 44, 49–50
  inspection 386, 389–390
  reply 33
  request 33
  spoofing 381, 389
  table 33
administrative logon 123
Advanced Encryption Standard (AES) 430
advanced packet processing performance 60
advanced routing 59
AES (Advanced Encryption Standard) 430
AF (assured forwarding) 345
agent address 418
aggregate switching capacity 43
aggregation layer 46
air filter 87
airflow 85, 93, 95, 97
alarm relay contacts 528
Alarms 148
ALM (Alarm) 138
alternative facility 473
  name 471
alternative source address 471
analyzer 501
  VLAN 501
application-specific integrated circuit (ASIC) 43, 46, 51, 57, 59–60, 111
area 289
area border routers 289
ARP (Address Resolution Protocol) 33
  cache 33
  entries 44, 49–50
  inspection 386, 389–390
  reply 33
  request 33
  spoofing 381, 389
  table 33
AS (adaptive services)
  boundary routers 289
  confederation 312
  path 301
ASCII file 113, 166
ASIC (application-specific integrated circuit) 43, 46, 51, 57, 59–60, 111
assured forwarding (AF) 345
ATM MIB 411
authentication 370
  certificate 145
  failures 419
  profiles 399
  server 390
Authenticator Port Access Entity 390
automatic software upgrade 177
auto-MDX mode 208
autonegotiation 14, 208, 212, 214
  status 210
autonomous systems 301

B

BA (behavior aggregate) 341
  classifiers 337
backbone areas 289
backplane 72
  speed 43, 48–49
backup
  role 172–173
  router 141
  switch 186, 188, 190
BE (best effort) 345, 352
behavior aggregate (BA) 341

© Copyright IBM Corp. 2011. All rights reserved.

classifiers 337
Bellman-Ford algorithm 276
best effort (BE) 345, 352
BFD (Bidirectional Forward Detection) 55, 57
BGP (Border Gateway Protocol) 36, 60, 112, 275, 278
identifier 312
route reflection 311
standards 304
Bidirectional Forward Detection (BFD) 55, 57
blocked ports 260
blocking interface 260
boot 134
Border Gateway Protocol (BGP) 36, 60, 112, 275, 278
identifier 312
route reflection 311
standards 304
boundary routers 289
BPDU (Bridge Protocol Data Unit) 26, 249, 259
protection 28, 259
braided ring 179, 184
bridge 19, 373
Bridge Protocol Data Unit (BPDU) 26, 249, 259
protection 28, 259
bridging 19
broadcast 16, 19–20
address 320
data 245
domains 245

C

CA (certificate authority) 145
cabinet airflow 97
cabling 99
CAIDA cflowd utility 411
candidate configuration 111, 160, 166, 523
candidate files 167
capacity utilization 148
Carrier Sense Multiple Access/Collision Detection (CSMA/CD) 13
central authentication 370
certificate authority (CA) 145
cflowd utility 411
chassis
alarms 528
cooling 84
grounding 82
identifier 262
process 110
status LEDs 137
CIDR (Classless Inter-Domain Routing) 34, 301
circuit cross-connect (CCC) 372
CIST (common and internal spanning tree) 252
class 15, 370
of powered devices 15
usage profiles 411
class D addresses 321
class of service (COS) 333–334, 345, 349, 372
Classless Inter-Domain Routing (CIDR) 34, 301
class-of-service MIB 411
cleanup 494
clearance for maintenance 93, 95, 99
CLI (command-line interface) 122, 152, 173, 494, 496
level 154
modes 152
terminal 152
tools 149
coaxial cabling 5
code point alias 341
collect statistics for interfaces 411
collision detection 13
command completion 155
command-line interface (CLI) 122, 152, 173, 494, 496
level 154
modes 152
terminal 152
tools 149
commit command 161, 166
commit delay timer 416
commit synchronize 112
common and internal spanning tree (CIST) 252
common spanning tree (CST) 252
community name 450
CompactFlash card 139
compare configuration 167
changes 525
component LEDs 529
configuration
commit scripts 114
hierarchy 149
history 112
management 151, 410, 523
management MIB 410
mode 114, 140, 152, 160–161
configure exclusive 161
configure private 161
Configure tab 149
console 122
access 153
port 99, 122, 139, 200
context-sensitive help 159
control virtualization 58
cooling system 72, 83, 85, 88, 90
clearance 93, 95
copying files 494
COS (class of service) 333–334, 345, 349, 372
count 156
CPU
control packets 225
load 148
craft interface LEDs 528
CRC (cyclical redundancy check) 16
CSMA/CD (Carrier Sense Multiple Access/Collision Detection) 13
CST (common spanning tree) 252
customer support 151
cyclical redundancy check (CRC) 16

D

DAA 60
DAI (dynamic ARP inspection) 381

daisy-chained ring 178
dashboard tab 148
data center network simplification 64
Data Encryption Standard (DES) 431
data rate 48–49
data terminal equipment (DTE) 99
date and time settings 127
DB9 99
DB-9 serial port 122
DB-9 serial port connector 139
dc output voltages 69
default aliases 341
default factory configuration 496
default gateway 37
default IP address 124
default link
    mode 209
    parameters 214
default preference value 310
default route 37
    preference value 278
default router 139
default schedulers 352
default system log settings 464
default wattage 218
delete command 165
deleting files 495
denial of service (DoS) attacks 381, 390
dense port concentrator 52–53, 336
    slots 60
DES (Data Encryption Standard) 431
destination address 16
device
    management 410
    model 148
DHCP (Dynamic Host Configuration Protocol)
    option 82 382
    relay agent information option 382
    snooping 381, 385, 390
        database 381
diagnostic and troubleshooting protocols 374
diagnostic tools 504
Differentiated Services Code Point (DSCP) 336–337, 341, 350
    for IPv6 341, 350
Dijkstra algorithm 276, 287
direct routes 276
disabling an interface 215
disabling Power over Ethernet (PoE) 217
display command 157
displaying previous configurations 524
displaying system log message 488
distribution 112
DNS 374
    server 139, 141
domain name 139, 141
DoS (denial of service) attacks 381, 390
drop profiles 335
DSCP (Differentiated Services Code Point) 336–337, 341, 350
    for IPv6 341, 350
DTE (data terminal equipment) 99
dual Routing Engine 515
duplex 14, 208
    mismatch 534
DVMRP 112
dynamic ARP inspection (DAI) 381
Dynamic Host Configuration Protocol (DHCP)
    option 82 382
    relay agent information option 382
    snooping 381, 385, 390
        database 381
dynamic method 185
    nonprovisioning 184
dynamic VLAN 397

E

EAP (Extensible Authentication Protocol) 390
EBGP (external BGP) 301
edge interface 259
edge routers 242
EGP (Exterior Gateway Protocol) 301
egress 372
    firewall filter 372
EIA (Electronics Industry Association) 92
EIA-232 99
electromagnetic interference (EMI) 74
Electronics Industry Association (EIA) 92
Emacs 156
EMI (electromagnetic interference) 74
enabling Power over Ethernet (PoE) 217
encryption type 430
end-of-row deployment 184
endpoint location 263
enterprise-specific SNMP traps 410
environment notifications 419
Ethernet
    aggregation 54
    port 139
    protocol type 16
    router VPNs 56
    services 54
except command 157
expedited forwarding (EF) 345
explicit priority 469
export policy 281
Extended Power via MDI 263
extended Virtual Chassis configuration 171, 179
Extensible Authentication Protocol (EAP) 390
Extensible Stylesheet Language Transformation (XSLT) 114
Exterior Gateway Protocol (EGP) 301
external BGP (EBGP) 301
    multihop sessions 309
Exx 48 Port 100FX/1000B-X Line Card 45
Exx 48 Port 1GbE RJ-45 Line Card 45
Exx 8 Port 10GbE Line Card 46
EZSetup 123, 125, 135
    script 117

F

facility 467
fan condition 148
fan tray 84, 87
fast reroute 57
Fast Reroute (FRR)
    MPLS 59
fault management 410
FCS (Frame Check Sequence) 16
FIB (Forwarding Information Base) 61
fiber optic cabling 8
field-replaceable unit (FRU) 68, 70
filter duplicate SNMP requests 416
filtering duplicate 416
find command 158
firewall filter 371, 411
    policies 411
firewall-based analyzer 501
flapping 176
flash usage 148
flexible PIC concentrator 213
flow control 208, 214
forging 381
form factor 43, 48–49
forwarding classes 335, 345
Forwarding Information Base (FIB) 61
forwarding state 260
forwarding tables 276
frame 16
Frame Check Sequence (FCS) 16
frame forwarding 20
FreeBSD UNIX operating system 109
FRR (Fast Reroute)
    MPLS 59
FRU (field-replaceable unit) 68, 70
FTP 503
full duplex 14, 209
full mesh topology 118
full-duplex 214

G

GBIC (Gigabit Interface Converter) 11
Gigabit Ethernet (GbE) 41
Gigabit Interface Converter (GBIC) 11
graceful restart 59
Graceful Routing Engine Switchover (GRES) 42–44
graphical chassis 148
GRES (Graceful Routing Engine Switchover) 42–44
grounding cable 74
guest host 406

H

half duplex 14, 209
halt 493
hashing algorithm 26, 225
health status 148
help reference 160
help topic 159
high-density 42
hot-insertable 85
hot-removable 85
HTTP 113, 144
    access 146
    on all interfaces 146
HTTP over Secure Sockets Layer (HTTPS) 113, 144–145
    access 147
    certificate 147
    on all interfaces 147
HTTPS (HTTP over Secure Sockets Layer) 113, 144–145
    access 147
    certificate 147
    on all interfaces 147
hub 12
hyperterminal 140

I

IANA (Internet Assigned Numbers Authority) 321
IBGP (internal BGP) 277, 302
IBM Ethernet j-type routers 75
IBM Ethernet Router J02M 51
IBM Ethernet Router J06M 51
IBM Ethernet Router J11M 51
IBM Ethernet Switch J08E 40, 44, 119
IBM Ethernet Switch J16E 40, 44, 49, 119
IBM Ethernet Switch J48E 40–41, 119
IBM j-type e-series Ethernet switches 39
IBM j-type m-series Ethernet router 51
ICMP (Internet Control Message Protocol) 374
    traffic 376
idle 134–135, 137
IEC (International Electrotechnical Commission) 69, 73
IEEE (Institute of Electrical and Electronics Engineers) 390
IEEE 802.1 341
    code point 350
IEEE 802.1ad (DEI) code point 350
IEEE 802.1x
    authentication 401
IETF (Internet Engineering Task Force) 279, 291, 304, 371, 413
IETF standard 371
IGMP (Internet Group Management Protocol) 22, 322
    snooping 322
IGMPv1 322
IGMPv2 322
IGMPv3 322
IGP (Interior Gateway Protocol) 277–278, 287
inband management 124, 128–129, 368
inconsistency state 261
inet (Internet) 276
informs 440
ingress 372
ingress firewall filter 372
initial configuration 139
initial setup 139
initialization process 110
Institute of Electrical and Electronics Engineers (IEEE)

390
integrated Routing Engine 43
inter-AS routing 301
interface
alarm 528
description 215
disabling 215
process 110
routes 276
interface-specific traffic statistics 411
interface-stats 423
interframe gap 17
Interior Gateway Protocol (IGP) 277–278, 287
internal BGP (IBGP) 277, 302
internal routers 289
International Electrotechnical Commission (IEC) 69, 73
Internet Assigned Numbers Authority (IANA) 321
Internet Control Message Protocol (ICMP) 374
traffic 376
Internet Engineering Task Force (IETF) 279, 291, 304, 413
standard 371
Internet Group Management Protocol (IGMP) 22, 322
snooping 322
Internet Protocol Television (IPTV) 54–55
Internet RFCs 279, 291, 304
Internet Service Provider (ISP) 3
interpolated profile 356
interpreting system log message 480, 486
intra-AS routing 302
IP address 139, 141
classes 34
IP forwarding 33
IP precedence 341
IP routing 32, 275
IP source guard 382, 386
IP subnet 32
IPSec 60
IPTV (Internet Protocol Television) 54–55
IPv4 276
addressing 34
precedence code point 350
IPv6 276
addressing 36
IS-IS 112, 276
ISIS 60
iso (ISO) 276
ISP (Internet Service Provider) 3

J

J08E 44
J16E 44, 49
jumbo frames 17, 44, 48, 50, 246
Juniper Networks Steel-Belted RADIUS Enterprise Edition 394
JuniperMibs 457
Junos 45, 47, 51–52, 59, 61, 107–108, 225
architecture 109
CLI 113–114, 139, 143, 152
hierarchical configuration 162
operating system 61
routing protocol 112
routing tables 276
version 148
Junos software 410
file system 521
operating system 43, 46, 108
upgrade 514
JUNOScope 147
JUNOScript
API 113
client 147
XML scripting API 147
JUNOScript over SSL 146
J-Web 113
GUI 113, 143–144
interface 122, 124, 494
session 130
user interface 113

L

label-switched path (LSP) 276
LACP (Link Aggregation Control Protocol) 225, 229, 233
interface parameters 227
statistics 227, 229
LAG (link aggregation group) 171, 225, 233
LAN (local area network) 3
core 46
Layer 1 5
configuration 208
Layer 2 15, 43, 46, 223
mirroring 503
VPN 56, 58
Layer 3 31, 43, 46, 55, 273
mirroring 503
VPN 56, 58
LC connector 10
LCD panel 133, 495
menus 135–136
modes 134
LCD screen 148
licenses 151
line card 45
role 174
switch 187–188
link aggregation 26, 58, 225, 262
negotiation 232
Link Aggregation Control Protocol (LACP) 225, 229, 233
interface parameters 227
statistics 227, 229
link aggregation group (LAG) 171, 225, 233
Link Layer Discovery Protocol (LLDP) 29, 261, 263
Link Layer Discovery Protocol-Media Endpoint Discovery (LLDP-MED) 29, 41, 261, 263
capabilities 262
device class values 262
link mode 209
link settings configuration 214
LLDP (Link Layer Discovery Protocol) 29, 261, 263
LLDP-MED (Link Layer Discovery Protocol-Media End-

point Discovery) 29, 41, 261, 263
capabilities 262
device class values 262
load configuration 167
load rescue configuration 135
loading a configuration from a file 526
local area network (LAN) 3
core 46
local endpoint address 307
local engine ID 428
local preference 309
value 309
location independence 42
logical routers 57
loop protection 28, 260
loopback filter 374
loss priority 335
LSA packet 288
LSP (label-switched path) 276
lug specifications 74

M

MAC 177, 448
address 15, 17, 44, 48, 50, 239
flooding 382
limiting 381–382, 390
move limiting 381, 384
spoofing 382
table 148
MAC/PHY configuration status 262
Maintain tab 151
maintenance 134, 136–137
MAINTENANCE MENU 125
major alarm 138
Manage Access page 129
management
address 262
console 141
Ethernet interface 459
Ethernet port 201
IP address 129
port 100
process 110
services 374
virtualization 57
Management Information Base (MIB) 413, 420, 451
managing files 494
master role 172–173
master switch 186, 188
mastership election 187
algorithm 172
process 176
mastership priority 176, 187–188, 191, 204
setting 176
value 176
match 158
Maximum Frame Size 262
maximum number of aggregated interfaces 226
maximum PoE wattage 218
maximum transmission unit (MTU) 215, 262
discovery 306
MBGP (multiprotocol BGP) 301
MBone (multicast backbone) 320
MD5 (message-digest algorithm 5) 429
authentication 280
MDI (Medium Dependent Interface) 8
MDIX (Medium Dependent Interface Crossover) 8
MED (Media Endpoint Discovery) 308
Media Endpoint Discovery (MED) 308
Medium Dependent Interface (MDI) 8
Medium Dependent Interface Crossover (MDIX) 8
MEF 60
member ID 174–175, 188, 195
numbering 174
member interfaces 226
memory utilization 148
message-digest algorithm 5 (MD5) 429
authentication 280
messages
to a console 470
to a remote machine 470
to a user terminal 470
metric 281
value 294
value added 280
MF (multifield)
classification 346, 361
classifiers 337, 343
MGMT 100, 124
MIB (Management Information Base) 413, 420, 451
mid-plane 72
minimum BGP configuration 305
minor alarm 138
mirrored port 501
modular architecture 56
monitor interface 508
Monitor tab 150
monitor traffic 509
monitor VLAN 501
monitoring station 501
monitoring system commands 511
MPLS 55, 59–60, 276
experimental code point 350
Fast Reroute (FRR) 59
network virtualization 57
routing table 113
VPN 60
MPLS (Multiprotocol Label Switching)
experimental 341
MRP 60
MST (Master) 138
MSTI (multiple spanning tree instance) 252
MSTP (Multiple Spanning Tree Protocol) 26–27, 60, 247, 252
MTU (maximum transmission unit) 215, 262
discovery 306
multicast 16, 19–20, 60
address 320
data 245
forwarding cache 276

routing table 112
snooping 321
multicast backbone (MBone) 320
multicast reverse path forwarding (RPF) 276
multifield (MF)
classification 346, 361
classifiers 337, 343
multihop BGP session 309
multimode fiber 9
multiple active routes 277
multiple spanning tree instance (MSTI) 252
Multiple Spanning Tree Protocol (MSTP) 26–27, 60, 247, 252
multiple VLAN mode 25
multiprotocol BGP (MBGP) 301
Multiprotocol Label Switching (MPLS)
experimental 341

N

name of the router 141
native analyzer session 501
NC (Network Control) 345, 352
NETCONF API 113
network architecture 55
Network Control (NC) 345, 352
Network Interface Card (NIC) 19
network layers 64
network management 409
network management station (NMS) 412, 416, 441, 457
network policy 263
network security services 63
network segment 12
Network Time Protocol (NTP) 374
network topology 118
network utilities 498
network virtualization 55, 57
next hop 317
NIC (Network Interface Card) 19
NMS (network management system) 412, 416, 441, 457
nonclient peers 311
nonstop routing 59
nonvolitile sets 423
not-so-stubby areas (NSSA) 290
NSSA (not-so-stubby areas) 290
NTP (Network Time Protocol) 374
number of VLANs 44, 49–50

O

object identifier (OID) 413, 420, 424, 441
OEM (Original Equipment Manufacturer) 39
OID (object identifier) 413, 420, 424, 441
one modular software architecture 62, 108
one operating system 62, 108
one software release 62, 108
Open Shortest Path First (OSPF) 36, 112, 275–276, 278, 287, 376
areas 289
designated router 290
external metrics 290
passive interface 295
routers 289
routing algorithm 288
standards 290
timers 295
Open System Interconnect (OSI) 2
operating mode change 213
operational mode 114, 152–154
operational-mode capabilities 155
operations support system 410
optical cable 102
maintenance 102
optical ports 101
Organizationally Unique Identifier (OUI) 18
Original Equipment Manufacturer (OEM) 39
OSI (Open System Interconnect) 2
OSPF (Open Shortest Path First) 36, 112, 275–276, 278, 287, 376
areas 289
designated router 290
external metrics 290
passive interface 295
routers 289
routing algorithm 288
standards 290
timers 295
OSPG 60
OUI (Organizationally Unique Identifier) 18
out-of-band management 124, 128, 201, 368

P

packet
capture 152
classification 335, 337
flow 336
forwarding capacity 60
processing 56
Packet Forwarding Engine (PFE) 43, 46, 53, 109, 111, 173–174, 213
packet loss priority (PLP) 352
password 139
recovery 497
pay-as-you-grow scalability 42
PDU (protocol data unit) 447
PEM (privacy-enhanced mail) 145
per port limit (PPL) 14
performance management 410
PFE (Packet Forwarding Engine) 43, 46, 53, 109, 111, 173–174, 213
physical ports 208
pie charts 150
PIM (Protocol Independent Multicast) 60, 112
ping 152, 498
pipe command 156
PLP (packet loss priority) 352
PoE (Power over Ethernet) 14, 29, 41, 101, 208, 217
configuration 217
controller 220
disabling 217
enabling 217

interface 217, 220
policy 261
ports 69
power consumption 218
policers 335
policing 346, 361
policy-based mirroring 501
port 208
densities 43, 59
description 262
firewall filter 372
identifier 262
mirroring 499, 503
priority 225
security 381
status 148
port VLAN 262
configuration 239
port-based network access control and authentication 390
POSIX standards 478
POST (power-on self-test) 119
power consumption 220
power distribution module 52
power management mode 14, 217
Power over Ethernet (PoE) 14, 29, 41, 101, 208, 217
configuration 217
controller 220
disabling 217
enabling 217
interface 217, 220
policy 261
ports 69
power consumption 218
power priority 218
power specifications 68
Power via MDI 262
power-on self-test (POST) 119
PPL (per port limit) 14
preamble 16
pre-emption 176
prefix length 141
preprovisioned configuration 188
preprovisioning method 184
previous configurations 524
priority scheduling 353
privacy-enhanced mail (PEM) 145
protocol analyzer application 501
protocol data unit (PDU) 447
protocol family 226
protocol independent 373
Protocol Independent Multicast (PIM) 60, 112
protocol-timeouts 423
public-private key 145
PuTTy 140

Q

QoS (quality of service) 56–57, 59–60, 333
queues/port 44, 48, 50
quality of service (QoS) 56–57, 59–60, 333
queues/port 44, 48, 50

R

racks 91
RADIUS (remote authentication dial-in user service) 371, 394
accounting 393
client 395
server 402
Random Early Detection (RED) 334, 345, 352, 356
Rapid Spanning Tree Protocol (RSTP) 26–27, 60, 247, 249
RED (Random Early Detection) 334, 345, 352, 356
Redbooks Web site 537
Contact us xvi
remote authentication dial-in user service (RADIUS) 371, 394
client 393, 395
server 402
remote port mirroring 501
repeater 12
rescue configuration 495
restart 151, 492
system 135
returning to a previously committed configuration 524
returning to the most recently committed configuration 524
revert to factory default configuration 135
reverting 495
rewrite marking 361
rewrite rules 335, 349
RFCs 413, 452
RIB (Routing Information Base) 61
RIP (Routing Information Protocol) 36, 60, 112, 275–276, 278
standards 279
timers 282
update messages 281
RIPng 276
RJ45 connector 6
RMON events 419
rollback 161
command 166
files 167
root ports 260
root protection 28, 261
rotating hard disk 139
route preferences 277, 294
route table 36
routed VLAN interface (RVI) 245
router firewall filter 372
Routing Engine 42–44, 46–47, 52, 86, 109, 122, 134, 139, 371
active 173
integrated 43
kernel 110
module 49
Routing Information Base (RIB) 61
Routing Information Protocol (RIP) 36, 60, 112, 275–276, 278

standards 279
timers 282
update messages 281
routing instance 443
routing protocol 36, 278
databases 275
notifications 419
OSPF 374
process 110
routing socket 423
RPM 152
RS-232 99
serial port 122, 139
RSTP (Rapid Spanning Tree Protocol) 26–27, 60, 247, 249
run command 168
RVI (routed VLAN interface) 245

S

sample traffic 411
save configuration 166
saving a configuration to a file 526
saving the current configuration 523
SC connector 10
scheduler 335, 352, 361
buffer size 353
drop profile maps 354
Scrutinizer NetFlow 463
Secure Hash Algorithm (SHA) 430
secure management 145
Secure Shell (SSH) 123, 129, 369, 374, 503
public key string (DSA or RSA) 141
Secure Sockets Layer (SSL) 145
certificate 145–146
encryption 145
server certificate 145
secure web access 145–146
security management 410
security model 438
security name 438, 451
serial cable 99, 139
service virtualization 58
service-level agreement (SLA) 56
severity 467
level 468, 477
SFD (Start Frame Delimiter) 16
sFlow 458
collector 459
data 459
sFlow Analyzer 463
SFP+ transceivers 213
SFP+ uplink module 213
SHA (Secure Hash Algorithm) 430
shaping rate 353
shared media segment 12
Shielded Twisted-Pair (STP) 5
Shortest Path First (SPF) 170, 287–288
uplink module 171, 179, 194
show commands 504
Simple Network Management Protocol (SNMP) 111, 124, 129, 374, 409
agent 447
community 450
community string 417
informs 413, 440, 447
MIBs 410
process 111
read community 124
tracing 421
trap group 418
trap options 417
trap queuing 414
traps 413, 440
single mode fiber 9
single Routing Engine 515
single VLAN mode 25
single-rate tricolor marking 347
SLA (service-level agreement) 56
SNMP (Simple Network Management Protocol) 111, 124, 129, 374, 409
agent 447
community 450
community string 417
informs 413, 440, 447
MIBs 410
process 111
read community 124
tracing 421
trap group 418
trap options 417
trap queuing 414
traps 413, 440
SNMPv1 412
SNMPv2c 412
SNMPv3 412
authentication type 429
cmmunity 450
minimum configuration 426
trap notification 440
user 428
software 151
packaging 513
source address 16, 418
Spanning Tree Protocol (STP) 26, 247–248
speed 208
SPF (Shortest Path First) 170, 287
uplink module 171, 179, 194
spoofing 381, 390
SRE (Switch Fabric and Routing Engine) 47–48, 122, 124, 134
SSH (Secure Shell) 123, 129, 369, 374, 503
public key string (DSA or RSA) 141
SSL (Secure Sockets Layer) 145
certificate 145–146
encryption 145
server certificate 145
ST connector 10
staircase type profile 356
standard MIB 457
standard SNMP traps 410

Start Frame Delimiter (SFD) 16
stateful firewall 60
static routes 142, 275, 317
statistical sampling 501
status 134–135, 137
status LEDs 133, 137
STP (Shielded Twisted-Pair) 5
STP (Spanning Tree Protocol) 26, 247–248
string, adding 474
structured-data 469
structured-data format 469
stub areas 290
subagent 423, 459
subnet mask 35
summary screen 130
supplicant 390
  modes 392
switch control board 52
Switch Fabric 47
Switch Fabric and Routing Engine (SRE) 47–48, 122, 124, 134
switched segment 13
SYN flood DoS attack 369
synchronization 112
syntax
  checker 155
  checking 155
SYS (System) 138
system
  commands 511
  date and time 124
  halt 135
  information 148
  logging severity levels 414
  name 148, 262, 417
  snapshot 513
  warm and cold starts 419
system log 512
  messages 410, 464
system log message 488
  interpretation 480, 486

T

TACACS (Terminal Access Controller Access Control System) 371
TACACS+ (Terminal Access Controller Access Control System Plus) 371
tag 451
  list 443
tagged frames 24
tail drop profile 360
target parameters 444, 446
TCN (Topology Change Notification) 27
TCO (total cost of ownership) 182
Telco racks 91
Telnet 123, 129, 132, 153, 368, 503
temperature 148
  sensors 84
Terminal Access Controller Access Control System (TACACS) 371
Terminal Access Controller Access Control System Plus (TACACS+) 371
terminal emulator 123
text string 473
throttle interval 414
throttle threshold 414
throughput 48–49, 60
TIA-598C standard 9
time zone 124
timer 423
TLV (Type Length Value) 29, 261
top of rack 41
top of rack deployment 180
Topology Change Notification (TCN) 27
total cost of ownership (TCO) 182
trace files 512
trace operations 422
traceroute 152, 498
transceiver 11
transit areas 290
transmission rate 353
trap notification filter 441, 445
trap target address 442
trap target parameters 445
traps 440
Triple DES 431
Troubleshoot tab 152
troubleshooting tools 152
trunk 242
  interface 242
  port 242, 381
trusted DHCP server 381
trusted host 396
tunnel 321
tunnel-medium type 397, 399
Tunnel-Private-Group-ID 397, 399
tunnel-type 397, 399
twisted pair 101
twisted-pair cabling 5
twisted-pair crossover 7
two-color marking 347
two-rate tricolor marking 347
Type Length Value (TLV) 29, 261
type of service 287

U

UDP (User Datagram Protocol) 279
  port 443
unicast 16, 19
  address 320
  data 245
  frame 21–22
  routing table 112, 276
unique ID 225
UNIX-based operating system kernel 152
Unshielded Twisted-Pair (UTP) 5
untagged factory default VLAN 237
upgrading Junos software 514
uplink module 171
USB flash drive 139

USB serial adapter 139
user account 132, 141
  class 141
user creation 369
User Datagram Protocol (UDP) 279
  port 443
user-based security model (USM) 423
USM (user-based security model) 423
UTP (Unshielded Twisted-Pair) 5

V

VACM (view-based access control model) 423, 434
varbind-error 423
VCP (Virtual Chassis port) 135, 170
  cable 171, 191
  connector 102
  verifying the status 204
vendor-specific attribute (VSA) 393
verifying the status of Virtual Chassis port 204
view-based access control model (VACM) 423, 434
virtual backplane protocol 42
Virtual Chassis 41–43, 134, 169
  cabling 178
  configuration 192
  installation 184
  interconnect 102
  management 200
  member role 172
  operational monitoring 203
  switch 118, 224
Virtual Chassis port (VCP) 135, 170
  cable 171, 191
  connector 102
  verifying the status 204
Virtual Leased Line (VLL) 60
virtual local area network (VLAN) 23
Virtual Management Ethernet (VME) interface 201, 459
Virtual Private LAN Service (VPLS) 54, 58, 60, 245
  multihoming 59
virtual router 58
Virtual Router Redundancy Protocol (VRRP) 245, 324, 419
  multicast 275
virtualization 57
VLAN 23
  configuration 237
  firewall filter 372
  ID 239
VLAN (virtual local area network) 23
VLAN Spanning Tree Protocol (VSTP) 26–27, 247, 256
VLL (Virtual Leased Line) 60
VME (Virtual Management Ethernet) interface 201, 459
Voice over Internet Protocol (VoIP) 14, 56
VoIP (Voice over Internet Protocol) 14, 56
VPLS (Virtual Private LAN Service) 54, 58, 60, 245
  multihoming 59
VRF-lite 58
VRRP (Virtual Router Redundancy Protocol) 245, 324, 419
  multicast 275
VSA (vendor-specific attribute) 393
VSRP 60
VSTP (VLAN Spanning Tree Protocol) 26–27, 247, 256
VT100 terminal 156

W

WAN (wide area network) 4
WAP (Wireless Access Point) 208
wide area network (WAN) 4
Wireless Access Point (WAP) 208
Wireshark tool 462

X

XFP 170
  uplink module 171, 179, 194
XML 113
XSLT (Extensible Stylesheet Language Transformation) 114

Implementation of IBM j-type Ethernet Switches and Routers

(1.0” spine)
0.875”<->1.498”
460 <-> 788 pages

Implementation of IBM j-type Ethernet Switches and Routers

Introduction to Ethernet fundamentals and IBM j-type Ethernet switches and routers

Introduction to Junos software and the command-line interface

Guidance for planning, installation, and configuration

IBM j-type data center solutions running Junos software (from Juniper Networks) provide operational agility and efficiency, dramatically simplifying the network and delivering savings. With this solution, a network design has fewer devices, interconnections, and network tiers. Beyond the cost advantages, the design offers the following key benefits:

- Reduces latency
- Simplifies device management
- Delivers significant power, cooling, and space savings
- Eliminates multiple system failure points
- Performs pervasive security

The high-performance data center is built around IBM j-type e-series Ethernet switches, m-series routers, and s-series firewalls. This new family of powerful products helps to shape the next generation of dynamic infrastructure. IBM j-type e-series Ethernet switches meet escalating demands while controlling costs. IBM j-type m-series Ethernet routers are high-performance routers with powerful switching and security capabilities.

This IBM Redbooks publication targets IT professionals who sell, design, or administer IBM j-type networking solutions. It provides information about IBM j-type Ethernet switches and routers and includes the following topics:

- Introduction to Ethernet fundamentals and IBM j-type Ethernet switches and routers
- Initial hardware planning and configuration
- Other configuration topics including Virtual Chassis configuration, Layer 1, Layer 2, and Layer 3 configurations, and security features
- Network management features of Junos software and maintenance of the IBM j-type series hardware

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-7882-00 ISBN 0738435023