Targeting JNCIA

Study Guide for Exam JN0-201

By

Jeffrey Ringwelski, John Jacobs,
Tyler Wessels

This book is a work of non-fiction. Names and places have been changed to protect the privacy of all individuals. The events and situations are true.

ISBN: 1-4107-6496-6 (e-book)
ISBN: 1-4107-6495-8 (Paperback)

Library of Congress Control Number: 2003094108

This book is printed on acid free paper.

Printed in the United States of America
Bloomington, IN

1stBooks - rev. 07/28/03

Acknowledgements

The authors would like to thank, in no particular order of appreciation:

Evan Mennillo for his great work and continued attention to detail. Max Gabriel and Brandon Gaither for feedback and assistance.

All those who contributed, in their own way, to a project that ended up a bit more complex than originally thought.

Contents

CHAPTER 5: OSPF 89

CHAPTER 6: IS-IS 115

Figures

Chapter One

Preface

Targeting JNCIA

Overview

The explosive growth in packet switched computer networks in general, and the public Internet specifically, continues to fuel a need for skilled technicians and engineers at every enterprise and service provider level. Likewise, the need for certification testing to maintain a benchmark for measuring those skills continues to be a necessity. Juniper Networks® builds and services a line of routers designed around Internet Protocol (IP) that are upon the leading edge of packet switching network technology.

Since the inception of the Juniper M40 backbone router, it has become increasingly apparent to industry specialists that there is a new contender in a market that was previously dominated of Cisco Systems. As Juniper expands its visibility both at the core and edge of modern networks, it is becoming increasingly important for employers and employees alike to recognize the special skills and tasks required to work with these devices.

Introduction

The Juniper Networks Technical Certification Program (JNTCP) is a multi-tiered program that demonstrates a competence with Juniper Networks M and T-Series routers, JUNOS software, and general internetworking ability. Complete information on the certification program as well as the most up to date information regarding the exams can be found at www.juniper.net.

The current four tiers that constitute the JNTCP are:

Juniper Networks Certified Internet Associate (JNCIA) – A written exam administered by Prometric (www.prometric.com) consisting of 60 multiple choice questions that ensure an essential foundation of basic knowledge necessary for troubleshooting and debugging a variety of routing issues involving Juniper Networks devices. A minimum of 70% is required to pass. This is the base level for Juniper certifications. The JNCIA is valid for 2 years and may be renewed by taking the current version of the test.

Juniper Networks Certified Internet Specialist (JNCIS) - Testing for this level of certification is made up of a 75 question, multiple choice written exam available from Prometric testing centers. A minimum of 70% correct is needed for a passing grade. Pass/Fail results and scoring are available immediate after the test. Certification is valid for 2 years. A valid JNCIS is required for taking the JNCIP practical exam. There is no prerequisite for attaining the JNCIS certification.

Juniper Networks Certified Internet Professional (JNCIP) – The JNCIP is the first of the two full-day practical examinations (the JNCIE is the second). Test candidates have 8 hours to configure and troubleshoot a variety of routing problems using current Juniper Networks technology. Pass/Fail results and scoring are available within ten business days after the test. Certification is valid for 2 years. A valid JNCIS is required for taking the JNCIP practical exam.

Juniper Networks Certified Internet Expert (JNCIE) – A valid JNCIP certification is a prerequisite to begin the JNCIE exam. The JNCIE is the second full day hands-on practical exam. It requires expertise in configuring and debugging a variety of complex routing situations utilizing Juniper Networks technology. Pass/Fail status is available within 10 business days of taking the exam. The JNCIE is valid for 2 years and may be re-certified by taking the current JNCIS written exam.

Objective

The focus behind *Targeting JNCIA* is to enable the reader to have a working understanding of TCP/IP routing and the Juniper platform sufficient to allow him or her to pass the Juniper Networks Certified Internet Associate (JNCIA) certification written test. Before beginning this book, the reader should have a basic knowledge of TCP/IP and routing concepts. This includes: the OSI model, routing versus routed protocols, IP addressing and subnetting.

For a complete understanding of command and configuration syntax, access to a working M-series router with JUNOS is recommended. However, it is understood that access to such devices is a luxury that is not commonly available. As such, relevant examples of the output that can be expected from specific commands as well as configuration sections to help clarify potentially confusing material have been included.

This book is not designed to be network design handbook. Rather it is a study guide and may be used as a quick reference. As your knowledge of networking increases, it will become evident to you that many of the concepts and protocols presented here are much more complex and contain many additional features and caveats. Some of the more academic information has intentionally been omitted to ensure proper focus on the facts pertinent to passing the JNCIA examination. Study material for those interested in detailed discourse on protocols is noted at the end of every chapter by listing the Request for Comments (RFC) whitepapers that are stored at the IETF (www.ietf.org). Additionally, there are a number of detailed network design guides available for those planning on furthering their knowledge.

Using this Book

The primary objective of this book is to adequately prepare you for the JNCIA exam and get you on your way to achieving technical certification. *Targeting JNCIA* is broken into a number of sections to allow for ease of locating information.

The beginning section includes this preface, along with a refresher on IP sub-netting, OSI model layers, and overview of the Internet. The next section deals with the Hardware of Juniper Networks various devices, highlighting the similarities and differences between different platforms. The routing sub-section includes information on the major routing protocols and routing concepts which will be tested on the JNCIA exam: RIP, OSPF, IS-IS, BGP, MPLS, and Multicast. Included at the end of every chapter you will find a listing of additional reference material that, while it may not be covered on the test, may increase your understanding of the fundamental concepts previously covered.

To reiterate, *Targeting JNCIA* should not be thought of as a comprehensive guide to all things IP, rather it is a resource to prepare you for taking and passing the Juniper certification exam.

We have used different fonts in an attempt to prevent confusion about which sections of text represent router output, input, configuration sections, terms, and the like. Sample output and configuration sections will appear similar to:

```
jncia@my.router>
```

All configuration statements and router commands are listed so that optional parameters are enclosed in <angle brackets>.

```
        jncia@my.router> show interface <interface-name>
```

Tokens that should be replaced with actual interface or address information are *italicized*. In the above example, “interface-name” is optional for this command to function. If it is desired to view a specific interface, the actual name of that interface must be typed in to replace the string *interface-name*. Configuration commands that require an entry, but have multiple choices, are enclosed in [square brackets]. Be aware that JUNOS configuration levels are also noted with [square brackets]. In the example below, the level must be entered. Either a value of ‘1’ or ‘2’ can be configured. The interface name is also required, and data must be entered by the user to specify.

```
[edit protocols isis]
jncia@my.router#   set   interface   interface-name   level
[1|2] [enable|disable]
```

Again, *italicized words or strings* must be replaced with user data for the command to function. Optional commands are enclosed in <angled brackets>. Lastly, when data must be entered, but there are multiple values from which to choose, all appropriate values are enclosed in [square brackets] and separated with the pipe “|” character.

It is also important to note that most commands within JUNOS can have tags added to the end of them (‘brief’, ‘detail’, ‘extensive’). Not all possible command outputs are addressed when listed in this book. Most commands are truncated to maintain focus on the key points necessary for the exam.

Additionally, most configuration snippets will appear in their JUNOS tree format. Key terms and concepts will be *italicized* the first time they appear. As with most telecommunications guides, there is a considerable number of acronyms used throughout. Acronyms will be expanded the first time they are encountered and will be followed by their common abbreviation. We have included a glossary of terms to alleviate problems.

Finally, a number of diagrams are including using graphic icons to represent nodes and devices. The icons are standard router and switch representations and should be easy to recognize and understand.

In general, physical device connections are represented with a solid line between devices and logical connections (routing protocol adjacencies, for example) are illustrated via a dashed line between network elements.

We apologize that the font size is often altered with respect to configuration and output display. This was necessary to ensure all pertinent information stayed on the same line as frequently as possible and was therefore more legible.

Basics

The following sections are designed to give some background and a light refresher for material that forms the foundation for moving on to more advanced routing concepts and practices. These sections are not intended to teach someone who is totally unfamiliar with TCP/IP routing subject matter, but may clear up some trouble points for those who are just beginning or perhaps haven't been in practice for awhile. In particular, be certain you are comfortable with the idea of *classless inter-domain routing* (CIDR) and subnetting, especially being able to derive subnets and masks as it *will* play an important part in the JNCIA.

The Internet

The Internet is a worldwide collection of private and public computer networks which are interconnected to each other via a system of telecommunication service providers who transport data between end points. As with most consumer services, the nuts and bolts of the Internet are largely transparent to the average user. However, to those who must maintain and service sections of this huge "network's network", an understanding of these inner workings are critical. In order to set some type of standard on networks and interconnection that would ensure interoperability without compromising innovation, the *International Organization for Standardization* (ISO) developed a model for computer networking called the *Open Systems Interconnect* (OSI) reference model. The OSI model is broken down into 7 levels:

Layer 7 : Application
Layer 6 : Presentation
Layer 5 : Session
Layer 4 : Transport
Layer 3 : Network
Layer 2 : Data Link
Layer 1 : Physical

Table 1.1 OSI Model

The OSI model describes the flow of data in a network, from the lowest layer consisting of pulses on cables up to the highest layer containing the end user's software application. Data going to and from the network is passed layer to layer. Each layer is able to communicate with the layer immediately above and below it. Every one is written as an efficient, streamlined software component. When two computers communicate on a network, the software at each layer on one host is communicating with the same layer on the other. For example, the Application layer of one computer communicates with the Application layer of another. The Application layer on either host has no regard for how data actually passes through the lower layers of the other, nor does it have any control in how those lower layers behave. In a sense, the lower layers are transparent when communicating to another host.

While knowledge of the OSI model and the functions of its groups will help you to better understand TCP/IP and networking in general, the JNCIA examination utilizes very practical test questions. The main job of a router is to deliver data to physically diverse, logically grouped devices. This is the domain of the Network layer, the third level of the model. As such, the exam and this book concentrate upon the bottom three layers with a primary focus on layer-3.

IP Addressing

Internet Protocol (IP) is the de facto system for exchanging packets of data between nodes on the public Internet. This protocol defines the rules that must be followed for end hosts to communicate successfully using the network. Everything from email and streaming audio to web pages and voice traffic is "packetized" and moved from a source to a destination using IP. Digital bits are the basis for digital network communication. A bit can be either on or off, meaning this particular bit has a value of 1 or 0. Expanding this will illustrate the logical progression:

# bits	Possible Combinations
1	0 or 1
2	00 or 01 or 10 or 11
3	000 or 001 or 010 or 011 or 100 or 101 or 110 or 111

Table 1.2 Binary Bit Combinations

You can see that the combinations begin to add up rather quickly. This exponential rise is the basis for a common rule in addressing called 2^n. A group of eight bits is known as a *byte.* This gives a byte 256 possible combinations. The confusing part of this is that the first number is actually 0 (all bits 'off'). This means that these 256 combinations represent numerical values from 0 to 255.

When bits are organized in a byte, the bits on the left are high order bits while those on the right are low order. If we say that the positions begin on the right hand side, the value of each bits value is '2' to the power of its position if it is 'on' or zero if it is 'off'. Remember that binary numbering begins from 0. With that, the lowest order bit has a value of 2^0, or 1. The next higher bit has a value of 2^1, which is either 2 or 0. Next, 2^2, 4 or 0. Next 2^3, 8 or 0. Next $2^4 = 16$, $2^5 = 32$, $2^6 = 64$ 2^7 (the eighth bit, because numbering begins at 0!) is 128. So, if we layout the eight bits of a byte and assign them their values:

Bit Position	8	7	6	5	4	3	2	1
Value	128	64	32	16	8	4	2	1

Table 1.3 Binary Values

If we fully expand a byte in binary notation with bits being either 1 or 0 we can see how these values are derived. Because of this we can see that some of the possible combinations are:

00000001 = 1 00000010 = 2 00000011 = 3
00000100 = 4 00001000 = 8 00001101 = 13
00010000 = 16 00100000 = 32 00010111 = 23
01000000 = 64 10000000 = 128 11111111 = 255

IP addresses are in the format of 32 bits, or 4 bytes, usually given in familiar dotted decimal notation *x.x.x.x.* This gives a theoretical range of 0.0.0.0 to 255.255.255.255. Other times addresses will be given in expanded binary form. In such an expanded form, we would see 128.31.127.255 as:

10000000.00011111.01111111.11111111

IP Subnetting

Each unique node on an interconnected network requires a unique IP address to identify it. Each address has two parts: one which identifies a unique network and a second which identifies a unique host on that network. The concept of dividing hosts up into unique network identifiers is known as *subnetting*. Traditional 'classful' addressing defined groups of varying network size depending upon the value of the leading byte.

Address Class	**# Network Bits**	**# Hosts Bits**	**Decimal Address Range**
Class A	8 bits	24 bits	1-127
Class B	16 bits	16 bits	128-191
Class C	24 bits	8 bits	192-223

Table 1.4 Address Classes

Even to this day, IPv4 addresses are sometimes called by their historical class. This means that the class of an address can be identified simply by looking at the first octet. 12.123.240.101 belongs to the traditional space of a Class A address because the first octet (12) falls within the 1-127 range.

Certain addresses are reserved for specific uses. Two such types are broadcast and network addresses; when the host segment is all 'on' or all 'off', the binary equivalent of having all 1's or all 0's. The lowest address is considered the network and highest is the broadcast. This is what is frequently referred to as '2^n-2', the common formula for determining the useful number of addresses for a given power.

Keeping in mind that the number of hosts is determined by subtracting two addresses (2^n-2), one for the network and the other for broadcast, using the old Class A, B, and C addressing scheme the Internet could support the following:

- 127 Class A networks that could include up to 16,777,214 hosts each
- Plus 65,000 Class B networks that could include up to 65,534 hosts each
- Plus over 2 million Class C networks that could include up to 254 hosts each

Because Internet addresses were generally only assigned in these three sizes, there were a lot of ‘wasted’ addresses. For example, if you needed 10 addresses for your network you would be assigned the smallest address available (Class C). However, that still meant 244 unused addresses. As the number of nodes on the Internet continued to grow, it became apparent that the classful addressing scheme was not going to scale. While the Internet faced an “address crunch”, less than 10% of assigned addresses were actively used. To help combat the encroaching exhaustion of address space and simultaneously eliminate some of the waste, a system known as classless inter-domain routing (CIDR) was devised. CIDR was developed to be an efficient replacement for the old process of assigning Class A, B and C addresses with a generalized network “prefix”. Instead of being limited to network identifiers (or “prefixes”) of 8, 16 or 24 bits that corresponded to bytes, CIDR uses a *variable length subnet mask* (VLSM) to adjust the size of the sub-network. Thus, blocks of addresses can be assigned to a single host or networks as small as 2 hosts to those with over 1 million. This allows for address assignments that much more closely fit an organization's specific needs.

A subnet mask can be thought of as just that, a mask which is laid over the IP address to determine the network and host information. For this reason, IPv4 masks are also 32 bits, normally given in 4-byte x.x.x.x notation. Subnet masks are started from the left hand side and continue right. Any bit that is part of the mask is used to determine the network portion of the address, and is turned ‘on’ (1). Because the network ID must be continuous, the subnet mask ends when a bit on the right is turned ‘off’ (0). All subsequent bits are then ‘off’ as well. For example, a default Class C subnet mask includes the first twenty-four bits of the network section. This translates to the first three octets being ‘on’.

11111111.11111111.11111111.00000000 binary
or 255.255.255.0 in decimal notation.

When a subnet mask is used with an IP address, bits that fall to the right of the network section are the bits usable for host IDs. It is often easier to understand the bit pattern interaction of addresses and masks when they are expanded into binary notation. The binary equivalent of the IP address 192.168.12.100 is:

11000000.10101000.00001100.01100100

If we use a /24 subnet mask on this address:

11000000.10101000.00001100.01100100 address
11111111.11111111.11111111.00000000 mask
Network. Network . Network . Host

We can see from the above that the host portion is made up of the last octet. So, the .100 portion of the IP address identifies the unique host.

A CIDR address includes the standard 32-bit IP address and also information in the form of a subnet mask to determine how many bits are used for the network prefix. For example, in the CIDR address 10.10.134.192/30, the "/30" indicates the first 30 bits are used to identify the unique network and the remaining 2 bits describe the unique host. This is also sometimes noted, especially in writing router configurations, as a decimal subnet mask of 255.255.255.252. Looking at the last octet we can see that the rightmost two bits have not been included.

11111111.11111111.11111111.11111100

CIDR continues to follow the 2^n-2 rule in that there are 2 addresses subtracted for network and broadcast identifiers. There is one exception, any address with a /32 address denotes a single specific host. There is no network or broadcast address associated with a /32. Below is a partial listing of available VLSM CIDR addresses. In each case, the number of hosts available corresponds to 2^x - 2, where x is the length of the address in bits (32) minus the number of bits in the subnet mask (the number behind the /). So, 10.10.134.192/28 has 32-28= 4 bits. X=4 gives us 2^4-2=14 unique hosts available.

CIDR Block Prefix	**# Equivalent Class C**	**# of Host Addresses (2$^{\wedge n}$-2)**
/30	1/128th of a Class C	2 hosts
/27	1/8th of a Class C	30 hosts
/26	1/4th of a Class C	62 hosts
/25	1/2 of a Class C	126 hosts
/24	1 Class C	254 hosts
/23	2 Class C	510 hosts
/22	4 Class C	1,022 hosts
/21	8 Class C	2,046 hosts
/20	16 Class C	4,094 hosts
/19	32 Class C	8,190 hosts
/18	64 Class C	16,382 hosts
/17	128 Class C	32,766 hosts
/16	256 Class C = 1Class B	65,534 hosts
/15	512 Class C	131,070 hosts
/14	1,024 Class C	262,142 hosts
/13	2,048 Class C	524,286 hosts

Table 1.5 CIDR Blocks

Despite the advent of CIDR, concerns about the exhaustion of IP address space have become more prevalent recently simply given the number of nodes residing on the Internet. The present scheme of addressing with 4 bytes is known as IPv4 (version 4). IPv4 is over 20 years old, and has presided over a period of unprecedented, explosive growth. A new system, called IPv6 or Ipng (Next Generation) attempts to alleviate concerns of IPv4, most notably IP address exhaustion. Where an IPv4 address is 32 bits, an IPv6 address is 128 bits in length. This allows for a gargantuan number of addresses, over 340 unidecillion, or 3.4x10^{38} – over 1 billion addresses per person on the planet.

IPv6 is *not* a requirement for the JNCIA.

Key Points

- The OSI networking model contains 7 Layers:
    - Layer 7: Application
    - Layer 6: Presentation
    - Layer 5: Session
    - Layer 4: Transport
    - Layer 3: Network
    - Layer 2: Data Link
    - Layer 1: Physical
- IP addresses consist of 32 bits separated into network portion and a host portion.
- A subnet mask is used to determine the break between network and host subsections of an IP address.
- VLSM allows for conservation of address space by allowing the subnet mask to allocate bits across the entire range of the IP address.
- An address where all the host bits are set to 1 is the broadcast.
- An address where all the host bits are set to 0 is the network.
- There are 2^n-2 addresses in a CIDR block, where N is the number of host bits.

RFC

For more detailed technical information search for the following:

- RFC 1517: Applicability Statement for the Implementation of CIDR
- RFC 1518: An Architecture for IP Address Allocation with CIDR
- RFC 1519: CIDR: An Address Assignment and Aggregation Strategy
- RFC 1520: Exchanging Routing Information Across Provider Boundaries in the CIDR Environment
- RFC 1631: IP Network Address Translator (NAT)
- RFC 1812: Ipv4 Router Requirements
- RFC 1878: Ipv4Variable Length Subnet Table
- RFC 1918: Address Allocation for Private Internets
- www.ipv6.org
- www.ietf.org

Chapter Two

Hardware

Chapter 2: Hardware

Overview

M-series routers have two major architectural components: the *Routing Engine* (RE) and the *Packet-Forwarding Engine* (PFE). The RE and PFE separate the control plane and the forwarding plane within the router. The RE contains routing protocol overhead and route table information. The PFE's primary function is the forwarding of production traffic given the information supplied by the RE. While operating independently, the RE and PFE communicate to each other over a 100 Mbs internal link (known as fxp1). This separation of the control and forwarding planes allows the RE to process control packets, such as routing updates, without negatively impacting the performance of the PFE or throughput of the router.

Introduction

This chapter will cover the hardware of Juniper Networks M-series routers. While there are a number of subtle differences between the platforms, Juniper has endeavored to make their routers have a common feel to them, regardless of model. This will no doubt become apparent when you review the software section, but also rings true for hardware. The most notable common thread for the M-series routing architecture is the separation of the control plane from the forwarding plane. This design allows the router to process routing updates without reducing its ability to maintain line-rate forwarding. By the end of this chapter you should be able to:

- ✓ Identify the different M-series routers.
- ✓ Recognize different hardware components and how they relate to the boot process.
- ✓ Identify the primary hardware components of each M-series router.
- ✓ Understand the Application Specific Integrated Circuit (ASIC) layout.
- ✓ Explain packet flow through the ASICs.

The JNCIA focuses upon the widely available M-series Juniper routers. Recently, Juniper has unveiled its next generation of core routers, the T-Series. However, as the current exam does not focus upon these particular routers, neither will this chapter.

M-series Routing Engine (RE)

The primary function of the RE is to maintain the routing tables and control the routing protocols. The RE is also responsible for all software processes that control interfaces, chassis components, system management, and user access to the router.

All routing protocol packets are sent directly to the routing engine for processing. The software processes are run separately so that the failure of one process doesn't affect the other processes. The advertisement, filtering, and modification of routes are handled by the RE according to the configured routing policy. The RE is responsible for building and maintaining multiple routing tables. It derives the active routes from each routing table and creates the *forwarding table*. The master forwarding table is located locally on the RE and a copy is sent to the PFE (via fxp1). The copy of the forwarding table on the PFE is the instance used to actually switch packets through the router. The RE has the ability to update the forwarding table that resides in the PFE without disrupting packet flow.

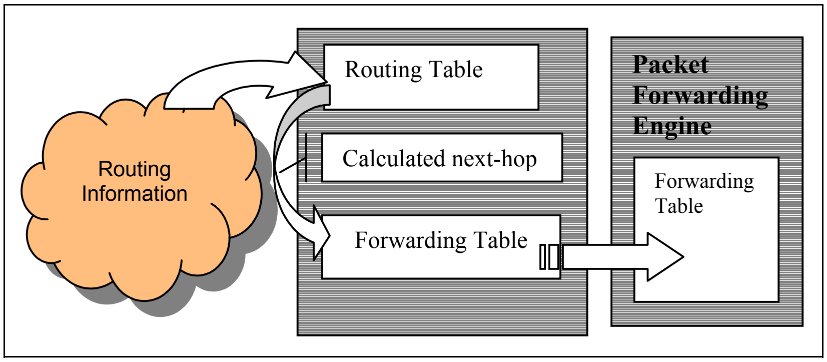


Figure 2.1: Flow and maintenance of the Routing and Forwarding Tables

The routing engine is the also the primary storage device for the router. Configuration files, JUNOS software, and microcode are stored and maintained in RE storage systems permitting local and remote upgrades.

The RE consists of a CPU, SDRAM, compact flash, rotating hard drive and a removable PCMCIA device. The CPU is a Pentium-class processor running JUNOS software. The SDRAM holds the routing table and forwarding table as well as other RE processes. Compact flash provides primary storage for JUNOS software images, microcode and two configuration files. The hard disk provides secondary storage for log files and memory dumps and an additional eight previous configuration files.

The boot source for an M-series router is as follows:

1) PCMCIA or ATA flash card (not often used)
2) Compact flash (also referred to as the non-rotating media)
3) Hard disk (also referred to as the rotating media)
4) Management Ethernet (network)

The RE controls all interfaces for out-of-band management access such as console ports, auxiliary (AUX) ports and the management Ethernet port. For more information on the software processes run on the RE, see Chapter 3, *JUNOS*.

M-series Packet-Forwarding Engine (PFE)

The primary responsibility of the PFE is to provide layer-2 and layer-3 packet switching. The PFE performs these functions through the use of *Application Specific Integrated Circuits* (ASICs). Each M-series router's PFE shares many of the same ASICs. The physical location of ASICs in the system varies between platforms, but the responsibilities and functions of the ASICs remain the same. Unlike the RE, which is a single component, the PFE is a distributed group of a number of hardware elements centered on optimizing packet forwarding. Unfortunately for those taking this exam, the group of components that make up the M-series PFE are not consistent across platforms. The larger, more robust, routers tend to have more individual, discrete components. The smaller boxes lean toward hardware consolidation to lower costs.

M-series PICs & FPCs

All media types (fiber, coax, UTP, etc.) require a physical connection to the router. The *Physical Interface Card* (PIC) is the first place a packet is received by the router and the last point it exits before going onto the transmission media. There are numerous types of PICs, varying by port speed, media type, port density, and so on. All M-series routers utilize them, but not every type of PIC can be used in every model of router.

For high-throughput routers like the M160, M40, and M20, PICs are arranged on removable *Flexible PIC Concentrators* (FPCs). On the smaller M5 and M10 routers, the FPCs are built into the *Forwarding Engine Board* (FEB). The major function of an FPC is to house PICs, shared memory for those PICs, *I/O Manager ASICs*, and *Packet Director ASICs*. These last two hardware components are discussed below.

M-series PFE ASICs

Application Specific Integrated Circuits (ASICs) are special chips designed to perform specific tasks. They are ideally suited for network devices as once a routine can be performed by a hardware chip, it is inherently faster than the same routine run on software. The hardware components that make up the PFE house the ASICs. The types of hardware components and location of the ASICs vary between M-series routers.

Each type of PIC is equipped with an ASIC that is designed to perform the media specific control functions for that type of interface. Encapsulation, de-capsulation, framing and checksums are some of the control functions provided by the media specific PIC ASICs.

The Packet Director ASIC is only utilized by M160 and M40e routers. The Packet Director ASIC's primary function is to distribute incoming packets to the I/O Manager ASICs and to distribute outgoing packets to the correct PIC.

The I/O Manager ASIC has 2 primary functions. The first is to divide incoming packets into 64-byte data cells (also called J-cells) and transfer the cells to the *Distributed Buffer Manager* (DBM) ASIC. The second function is to retrieve the 64-byte data cells from shared memory and reassemble the packet.

Each M-series router has 2 Distributed Buffer Manager ASICs. One Distributed Buffer Manager ASIC is responsible for managing and distributing the 64-byte data cells to shared memory banks that reside on the FPCs. The other is responsible for transferring outgoing packets to the correct FPC.

Route lookups are performed by the Internet Processor II ASIC using the forwarding table stored in RAM. This ASIC is located on different hardware components based upon the model of M-series router. The Internet Processor II ASIC is also responsible for transferring exception and control packets to the RE. Any packet that is required to be processed by the routing engine is considered to be an exception packet.

The below picture illustrates the sequential flow of a packet through the M40 and M160s ASICs. These ASICs make up the Packet Forwarding Engine. The Packet Director ASIC is only utilized by M40s and M160s.

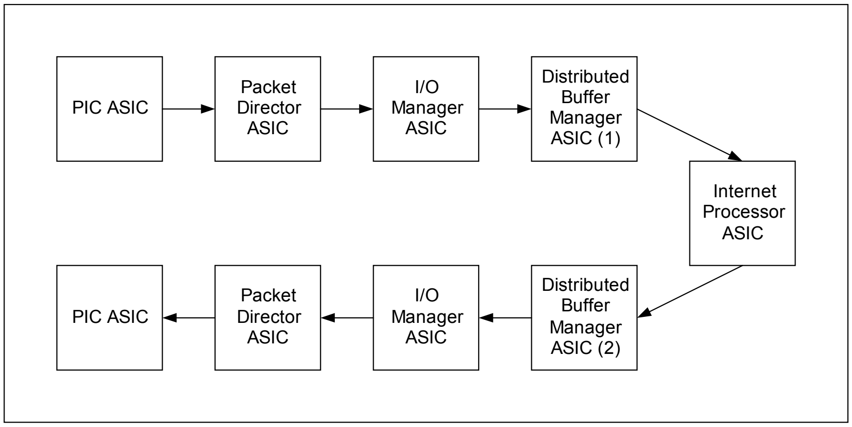


Figure 2.2 M160 & M40e packet flow (logical ASIC view)

The below picture illustrates the sequential flow of a packet through the M5/10 and M20 ASICs. These ASICs make up the Packet Forwarding Engine. Notice the lack of a Packet Director ASIC.

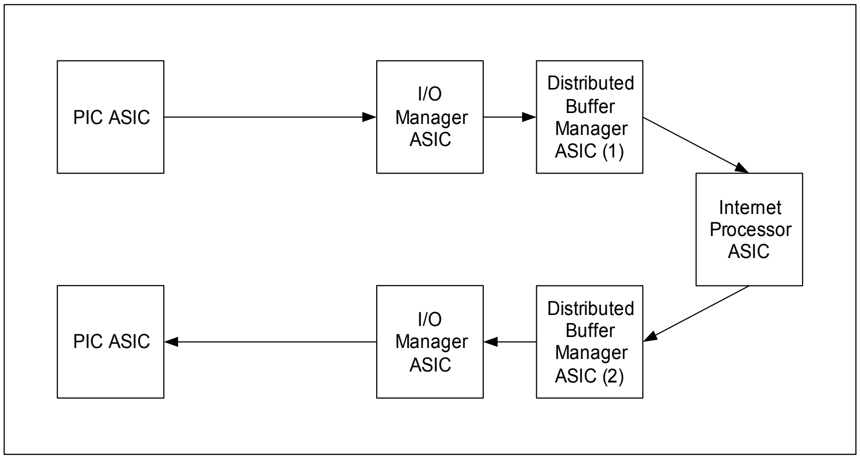


Figure 2.3 M5, M10, M20, and M40 packet flow (logical ASIC view)

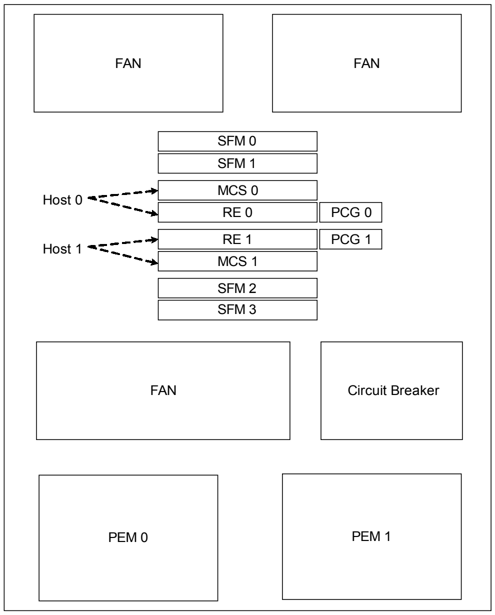
M160 Overview

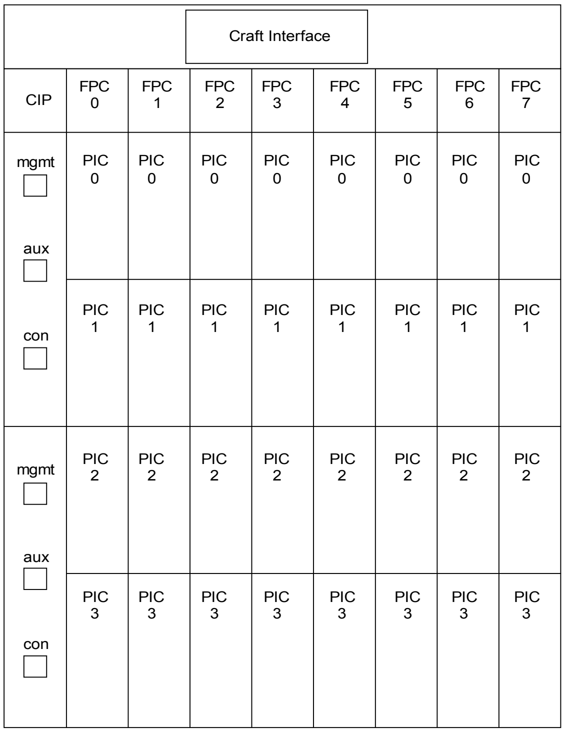
The M160 is an internet backbone class router offering high-speed SONET/SDH, ATM, and Gigabit Ethernet media types. This router is designed for large networks such as those used by *Internet Service Providers* (ISPs). The M160 has an aggregate throughput of 160 Gbps and can forward up to 3 Gbps at line-rate on each original FPC1 and up to 10 Gbps at line-rate on each next-generation FPC2.

M160 Chassis

The chassis is the structure that houses all the individual hardware components. The primary component is the *midplane*, which is located vertically towards the back of the chassis. Each component that is installed in the chassis connects to the midplane. The midplane is responsible for transferring packets from one component to another, distribution of power to each component, and signal connectivity that is used for monitoring and control of the entire system.

M160 Chassis Back View





M160 Chassis Front View

Figure 2.4 M160 Chassis View (Front and Rear)

M160 Flexible PIC Concentrator (FPC)

FPCs are inserted at the front of the chassis connecting to the chassis midplane and house various PICs. There are eight vertical FPC slots located at the front of the M160. The slots are numbered from left-to-right where the leftmost slot is 0; the rightmost slot is 7. Each allows up to four PICs to be installed. If a slot is not occupied by an FPC, or if an FPC is not fully populated with PICs, a blank cover must be used to allow proper airflow and cooling within the chassis.

There are three basic types of FPCs: *type 1, type 2, and OC-192*. Type 1 FPCs (FPC-1s) support such interfaces as single-port OC-12 and Gigabit Ethernet (GigE) PICs. Type 2 FPCs (FPC-2s) support higher speed PICs such as OC-48 and 2-port GigE. The last type is an OC-192. This model does not have four individual connectors for PICs, but rather the entire FPC is dedicated to a single OC-192 interface. Type 1 and OC-192 FPCs are produced in two forms: standard and enhanced. Enhanced cards have advanced QOS capability and an additional 2MB of RAM.

All FPCs are hot swappable. The chassis does not have to be powered down to remove or install an FPC. An FPC whose PICs are not carrying live traffic that is removed will cause slight forwarding latency while the shared memory is flushed. One configured with PICs carrying live traffic cannot be removed without causing a network outage and packet loss.

FPCs connect the PICs to the rest of the router, allowing packets entering a PIC to be forwarded across the midplane to the SFMs and ultimately to the destination port. Each FPC contains a shared memory pool and two types of ASICs: one Packet Director ASIC and up to four I/O Manager ASICs. The primary role of the Packet Director ASIC is to accept packets from the PICs installed on the FPC and prepare them to be passed on to the I/O Manager ASIC. The I/O Manager ASIC divides each packet into 64-byte memory blocks that will be stored across all FPC shared memory by the Distributed Buffer Manager ASICs (located on the SFMs).

M160 Switching & Forwarding Modules (SFMs)

The SFMs are located in the rear of the chassis and constitute the majority of the PFE. Up to four of these hot-swappable components can be installed to provide full packet forwarding capability. Removing an SFM does disrupt forwarding performance as the PFE reconfigures the distribution of packets to the remaining SFMs. At least one SFM must be online for the router to continue forwarding packets.

The primary functions of the SFM are route lookup, buffer management and switching packets to a destination FPC. Each SFM has an Internet Processor II ASIC that performs route lookups using the forwarding table that is stored locally in SRAM. The Distributed Buffer Manager ASIC also resides locally on the SFM and is responsible for allocating incoming (from an FPC) packets to the shared memory pool located on all FPCs. There is a second Distributed Buffer manager ASIC that is responsible for forwarding outgoing (outgoing from the SFM) packets to the FPCs. Another function of the Internet Processor II ASIC is to transfer control and exception packets to the microprocessor on the RE. Any errors detected by the SFM's microprocessor are sent to the routing engine in the form of a syslog message that describes the error.

M160 Packet Flow

The following steps walk through packet on a M160 router:

- Packets first enter the router via a PIC interface.
- They are then sent to the Packet Director ASIC on the FPC.
- The Packet Director ASIC distributes the packets in a round-robin fashion to the FPC's I/O Manager ASICs.
- The I/O Manager ASICs process the packet header and divide the packets into 64 byte cells, forwarding the cells through the midplane to the inbound Distributed Buffer Manager ASIC on the SFMs. Note that *Quality of Service* (QoS) queuing takes place within this ASIC.
- The Distributed Buffer Manager ASIC distributes the 64-byte cells throughout the shared memory banks of each FPC.
- The Internet Processor II ASIC on the SFM performs the lookup and makes a forwarding decision.
- The Internet Processor II ASIC notifies the outbound Distributed Buffer Manager (DBM) ASIC on the SFM of the forwarding decision.
- The outbound DBM ASIC forwards the notification to the I/O Manager ASIC of the FPC that houses the outgoing PIC.
- The I/O Manager ASIC retrieves the 64-byte cells from the shared memory banks and reassembles the packet with the results of the route lookup done by the Internet Processor II ASIC.
- The I/O Manager ASIC then forwards the reassembled packets to the FPCs Packet Director ASIC who forwards the packets to the correct outgoing PIC.
- The PIC finally transmits the packets out the appropriate interface.

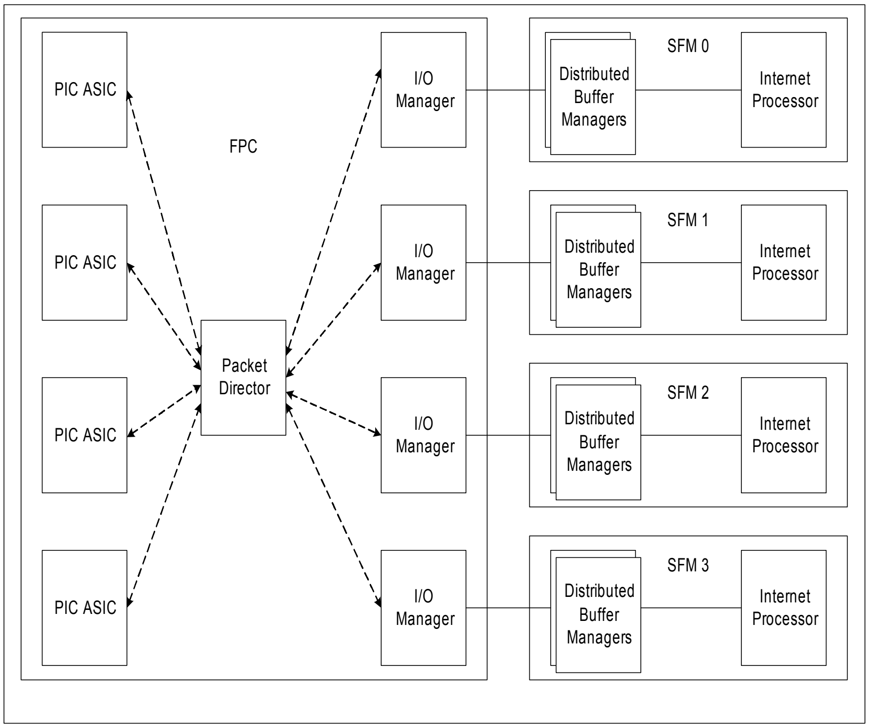


Figure 2.5 M160 Packet Flow (ASIC placement view)

Physical Interface Cards (PICs)

PICs are the connections to various network media types such as SONET/SDH, ATM, and Gigabit Ethernet. They transmit and receive network packets. PICs are responsible for the encapsulation, framing, and line speed signaling for its specific media type. PICs are hot swappable. If the PIC is carrying live traffic and is removed, packets will be dropped and a network outage will occur.

M160 PFE Clock Generators (PCGs)

The M160 router is configured with two PCGs located at the rear of the chassis. The RE dictates one as the primary and the other as the secondary. Each contains a 125 MHz system clock generator. The clock generator is used to provide timing and synchronization to the components of the PFE.

M160 Host Module

The host module actually refers to two separate hardware components that rely heavily upon the function of each other. An RE and *Miscellaneous Control Subsystem* (MCS) are physically separate, but function logically as a single unit. The router supports up to two host modules (two REs and two MCSs). An RE cannot operate without an adjacent MCS. If two host modules are installed, one is designated as active and the other as backup. Upon failure of the active host module, the backup module assumes the primary role.

M160 Miscellaneous Control Subsystem (MCS)

The MCS is installed at the rear of the chassis connecting to the midplane. As mentioned in the RE section, up to two MCSs can be installed. Each MCS installed requires an adjacent RE. The primary function of the MCS is to work with the RE in providing control and monitoring of the various router components. The MCS also provides the SONET clocking for the router while the PCGs provide system clocking.

The MCS monitors each component of the router for failures and alarms. Statistics from each component are collected by the MCS and then relayed to the RE, which will then generate the appropriate log message or alarm condition. For all components that have a master/backup relationship, the MCS dictates which of the two devices will be master.

M40e Overview

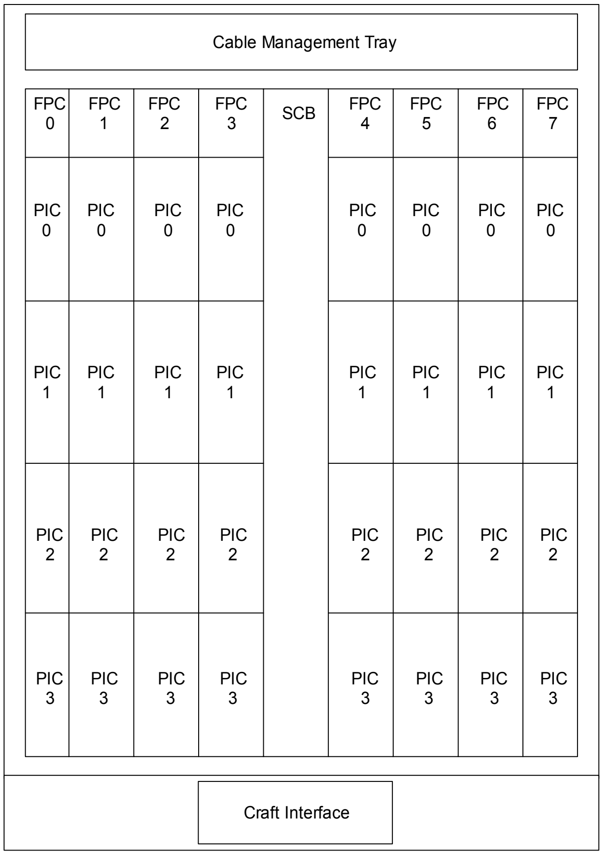
The M40e router is identical to the M160 except that it only supports up to two SFMs, unlike the M160 which supports up to four. If two SFMs are installed, one is active and the other is backup. This router's throughput capacity is subsequently reduced to 40 Gbps.

M40 Overview

The M40 has an aggregate throughput of 40 Gbps and can forward up to 3 Gbps at line-rate on each FPC. The M40 chassis has eight vertical slots for FPCs to be installed. The major architectural difference between the M40 and M40e/M160 is the PFE. The M40 PFE consists of a backplane, *System Control Board* (SCB), FPCs and PICs.

The same ASICs that are used on the M40e/M160 are also used on the M40. The only difference being the location of some of the ASICs. On the M40e/M160, the Internet Processor II ASICs are located on the SFMs. On the M40, the older Internet Processor I ASIC is located on the SCB (the Internet Processor I performs the same functions, but is not capable of some enhanced firewall features, as noted in the Policy chapter). On the M40e/M160, the Distributed Buffer Manager ASICs are also located on the SFMs. On the M40, they reside on the backplane.

M40 Front



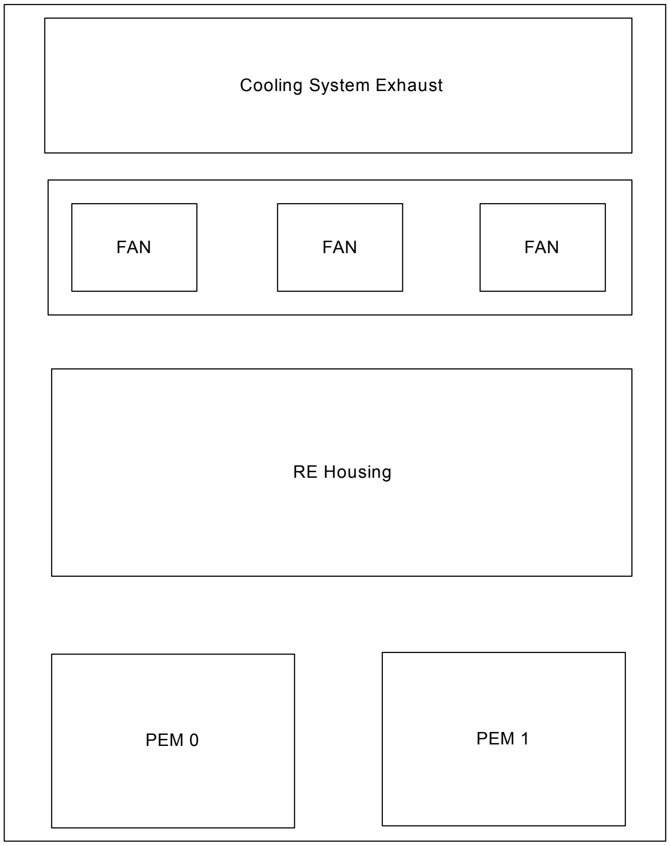


Figure 2.6 M40 Chassis View (Front and Rear)

M40 Backplane

The M40 backplane performs many of the same functions as the M40e/M160 midplane. The backplane is part of the PFE and performs 3 major tasks: power distribution, signal connectivity to the various router components, and housing of the Distributed Buffer Manager ASICs, which manage the shared memory on the FPCs.

M40 System Control Board (SCB)

The SCB connects to the backplane from the center vertical slot at the front of the chassis. It is part of the PFE and performs 4 major functions:

route lookups, system component monitoring, exception and control packet forwarding, and FPC control.

The Internet Processor I ASIC resides on the SCB and is responsible for performing route lookups. They are performed using the forwarding table that is stored on the SCB's *synchronous SRAM* (SSRAM). Similar to the MCS on the M40e/M160 platform, the SCB is responsible for monitoring the various router components for alarms and failure conditions. It also collects component statistics and relays this information to the RE, where the appropriate log message is generated or alarm condition is triggered. In addition to the standard component monitoring, the SCB has the ability to initiate an automatic reset of an FPC should such a problem or error arise.

M40 Packet Flow

The packet flow through the M40 is identical in theory to the packet flow through the M40e/M160 with regard to the ASICs. The various ASICs that make up the PFE are located on different components of the router, as noted, but the type and order of packet flow through the ASICs remains unchanged.

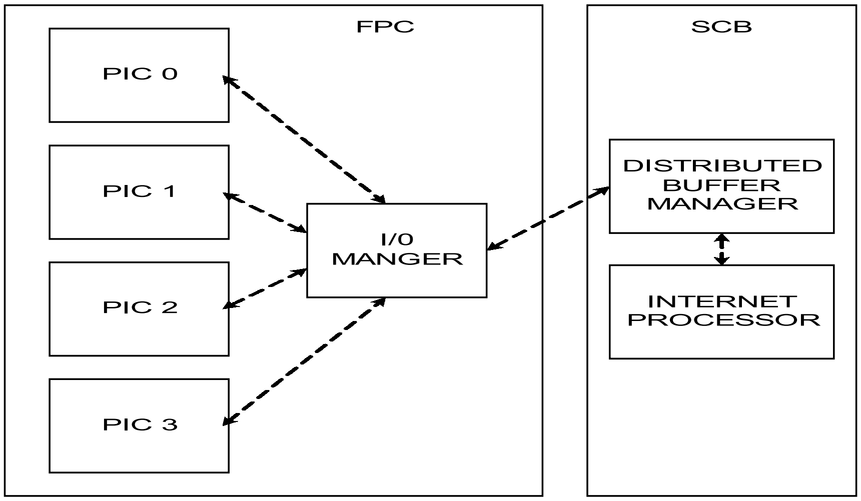


Figure 2.7 M40 packet flow (ASIC placement view)

M20 Overview

The M20s aggregate throughput is 20 Gbps and forward at line rate up to 3 Gbps on a single FPC. The chassis supports up to 4 horizontal FPCs that are installed at the front of the chassis connecting to the backplane.

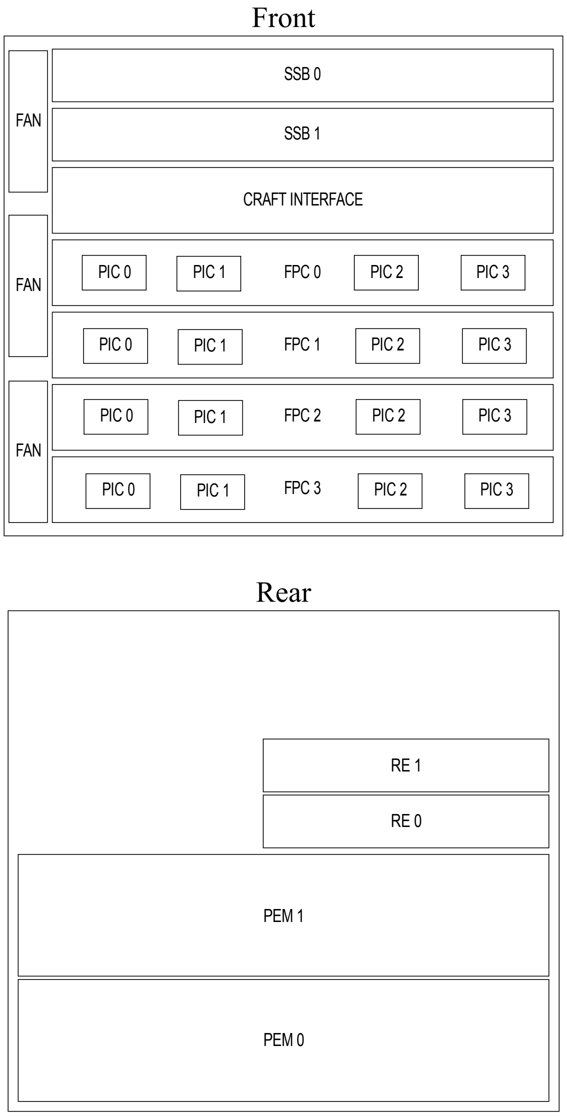


Figure 2.8 M20 Chassis View (Front and Rear)

M20 Packet Forwarding Engine

The M20 PFE consists of 4 components: the midplane, *system and switch board* (SSB), FPCs and PICs. The midplane forms the rear of the card cage where the FPCs and SSBs are connected and is responsible for power distribution and signal connectivity.

The SSB installs horizontally at the front of the chassis connecting to the midplane and houses the Internet Processor II ASIC and Distributed Buffer Manager ASICs. The SSB is responsible for much of the packet forwarding and overall system control. Some system control functions such as component monitoring and statistics collecting take place on the SSB. It monitors and collects statistics about alarm and error conditions of each of the router components. The system data that the SSB collects is passed on to the RE where the appropriate log message or alarm state will be set. The SSB also has the ability to reset FPCs if an alarm or error state is detecting that warrants a reset.

FPCs

The major difference in M20 FPCs is in the number of I/O Manager ASICs and the lack of a Packet Director ASIC. Because M20 FPCs only have a single I/O Manager ASIC, there is no need for a Packet Director ASIC whose purpose is to distribute packets to multiple I/O Managers. The FPCs perform the same functions as the FPCs from other M-series routers.

M-5/M-10 Overview

The major difference between the M5 and M10 routers is the number of PICs each supports. The M5 router supports up to four PICs while the M10 supports up to eight. The aggregate throughput of both is 6.4 Gbps. The M5 can forward up to 3 Gbps at line-rate for any combination of PICs, the M10 can forward up to 6 Gbps at line-rate for any combination of PICs.

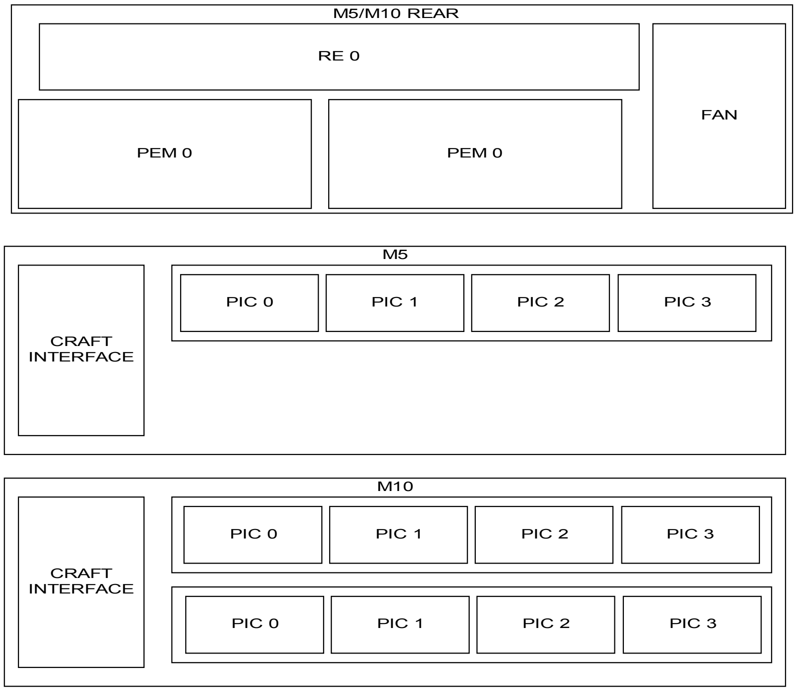


Figure 2.9 M5/M10 Chassis View (Front and Rear)

M5/M10 PFE

The PFE consists of 3 major components: the midplane, Forwarding Engine Board (FEB), and PICs. The midplane occupies the center of the router chassis where the FEB, PICs and other components connect. The midplane provides power distribution and signal connectivity.

The FEB is located at the rear of the chassis above the power supplies. It houses the Internet Processor II ASIC and two Distributed Buffer Manager ASICs. The Internet Processor II ASIC performs route lookups using the forwarding table that is stored in SSRAM on the FEB. It is also responsible for transferring exception and control packets to the RE for appropriate log message and alarm condition creation.

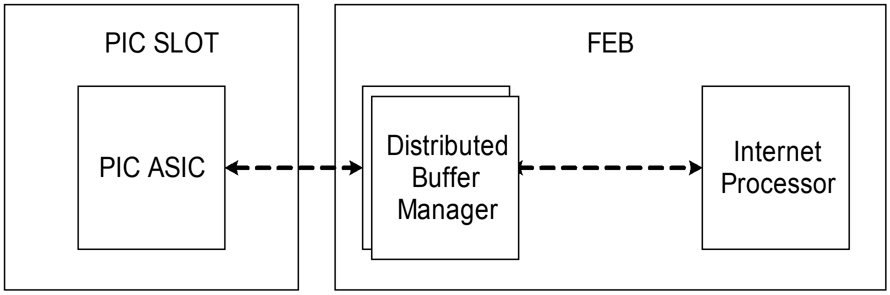


Figure 2.10 M5/M10 packet flow (ASIC placement view)

Key Points

- The two major components of every M-Series router are:
  - Routing Engine
  - Packet Forwarding Engine
- JUNOS software is kept on compact flash, a back-up is stored on the hard disk.
- The boot series for M-series routers is:
  1) PCMCIA or ATA flash card (not often used)
  2) Compact flash (also referred to as the non-rotating media)
  3) Hard disk (also referred to as the rotating media)
  4) Management Ethernet (network)
- The RE uses information from all protocols to build the Routing Table, which contains all destinations the router is aware of.
- The best next-hop addresses are used to build the Forwarding Table.
- The RE keeps a master copy of the Forwarding Table and sends a copy to the PFE over fxp.1.
- QoS queuing takes place on the I/O Manager ASIC.
- The PFE is made up of distributed components, and utilize ASICs to provide hardware forwarding functions. The key chips for each platform are noted below:

  M160/M40e ASICs:
  - PIC ASICs
  - 1 Packet Director per FPC
  - 4 I/O Managers per FPC
  - 2 Distributed Buffer Managers per SFM
  - 1 Internet Processor II ASIC per SFM

  M40:
  - PIC ASICs
  - 1 I/O Manager per FPC
  - 2 Distributed Buffer Managers on the backplane
  - 1 Internet Processor I ASIC on the SCB

  M20:
  - PIC ASICs
  - 1 I/O Manager per FPC
  - 2 Distributed Buffer Managers on the SSB
  - 1 Internet Processor II ASIC on the SSB

M5/M10:

- PIC ASICs
- 2 Distributed Buffer Manager ASICs on the FEB
- 1 Internet Processor II ASIC on the FEB

➢ Packet Flow for the M160/M40e:

1) Packets first enter the router via a PIC interface.
2) They are then sent to the Packet Director ASIC on the FPC.
3) The Packet Director ASIC distributes the packets in a round-robin fashion to the FPC's I/O Manager ASICs.
4) The I/O Manager ASICs process the packet header and divide the packets into 64 byte cells, forwarding the cells through the midplane to the inbound Distributed Buffer Manager ASIC on the SFMs. Note that *Quality of Service* (QoS) queuing takes place within this ASIC.
5) The Distributed Buffer Manager ASIC distributes the 64-byte cells throughout the shared memory banks of each FPC.
6) The Internet Processor II ASIC on the SFM performs the lookup and makes a forwarding decision.
7) The Internet Processor II ASIC notifies the outbound Distributed Buffer Manager (DBM) ASIC on the SFM of the forwarding decision.
8) The outbound DBM ASIC forwards the notification to the I/O Manager ASIC of the FPC that houses the outgoing PIC.
9) The I/O Manager ASIC retrieves the 64-byte cells from the shared memory banks and reassembles the packet with the results of the route lookup done by the Internet Processor II ASIC.
10) The I/O Manager ASIC then forwards the reassembled packets to the FPCs Packet Director ASIC who forwards the packets to the correct outgoing PIC.
11) The PIC transmits the packets out the appropriate interface.

➢ Packet Flow for the M40/M20:

(Note that the M20 has an Internet Processor II and an SSB rather than the Internet Processor I and SCB of the M40.)

1) Packets first enter the router via a PIC interface.
2) They are then sent to the FPC I/O Manager ASIC.

3) The I/O Manager ASIC process the packet header and divide the packets into 64 byte cells, forwarding the cells to the Distributed Buffer Manager ASIC on the SCB. Note that *Quality of Service* (QoS) queuing takes place within this ASIC.
4) The Distributed Buffer Manager ASIC distributes the 64-byte cells throughout shared memory.
5) The Internet Processor I ASIC on the SCB performs the lookup and makes a forwarding decision.
6) The Internet Processor I ASIC notifies the Distributed Buffer Manager ASIC on the SCB of the forwarding decision.
7) The outbound DBM ASIC forwards the notification to the I/O Manager ASIC of the FPC that houses the outgoing PIC.
8) The I/O Manager ASIC retrieves the 64-byte cells from the shared memory banks and reassembles the packet with the results of the route lookup done by the Internet Processor I ASIC.
9) The I/O Manager ASIC then forwards the reassembled packets to the correct outgoing PIC.
10) The PIC transmits the packets out the appropriate interface.

- Packet Flow for the M5/M10:
    1) Packets first enter the router via a PIC interface.
    2) They are then sent to the Distributed Buffer Manager (DBM) ASIC on the FEB
    3) The Internet Processor II ASIC on the FEB performs the lookup and makes a forwarding decision.
    4) The Internet Processor II notifies the DBM on the FEB of the forwarding decision.
    5) The DBM ASIC then forwards the packets to the correct outgoing PIC.
    6) The PIC transmits the packets out the appropriate interface.

Chapter Three

JUNOS

Chapter 3: JUNOS

Overview

JUNOS is the common operating system that is run on all Juniper M-series routers. All processes that control the router run on a UNIX kernel. The *command-line interface* (CLI) is a shell process that parses and inputs all user commands to JUNOS. This command set controls all aspects of the hardware and routing instructions. By the end of this chapter you should understand and be able to define:

- ✓ The router boot sequence.
- ✓ Features of the CLI.
- ✓ Processes of JUNOS.
- ✓ The JUNOS configuration tree.
- ✓ How to edit the configuration file.
- ✓ How to view traceoption logfiles.
- ✓ How to view and identify configuration groupings.

Introduction

JUNOS is the brain of a Juniper router, without it not much can get done. It is precisely because of this necessity that it is often overlooked in favor of studying the hardware it interfaces with and the data it processes. There are a number of things that need to be noted about this otherwise transparent layer of code to best prepare for the exam.

All M-series routers run the same JUNOS code. There is no special revision for M160's that will not work on an M5. This is contrary to some vendors who make specific code for specific platforms. Indeed, Cisco IOS® has many different trains of code within a specific platform's IOS depending on the feature set desired. This means that while there are hardware specific commands that may differ due to the chassis, such as the absence of SFMs on the M5, the overall command set is the same. This translates into increased productivity for a technician who no longer needs to recall different command conventions.

The JUNOS software resides on the compact flash (often called the non-rotating media or RAM disk) on the routing engine. The backup/alternate copy is stored on the hard drive (or 'rotating media'). The current/active configuration and the three previous configurations are stored on the internal flash drive for quick access. Meanwhile, the six previous configurations are stored on the hard drive (numbered 4 through 9).

Boot Process

The boot process is fairly lengthy and displays a large amount of information on the console. One of the key factors to acknowledge is from where the software has booted. If not from flash (either removable PCMCIA or compact), the router will post a message upon login stating that it has booted from alternate media.

Upon start up, the router will attempt to find a useable copy of JUNOS. The boot sources for an M-series router are as follows:

1) PCMCIA or ATA flash card (not often used)
2) Compact flash (non-rotating media)
3) Hard disk
4) Network (Ethernet)

If any of the above hardware components are missing or the code stored therein is corrupt, the router will move down the list to the next candidate media. It is recommended by most support personnel that the flash card be removed from the router as it is not often accessed and for this reason may hold an outdated version of software and configuration. This will force the router to boot from the compact flash and subsequently the hard drive if problems are encountered.

Processes (Daemons)

For the most part, the individual processes (called *daemons* in the UNIX world) that make up the full feature set of JUNOS software run independently. Each of these processes runs in an individual memory space and can most often be halted or restarted without impacting others. Diagnostic or cosmetic output processes, such as the *SNMP daemon* (snmpd), could be stopped or restarted with little impact to the overall packet forwarding of the router. However, it is fairly obvious that when a daemon such as the *routing process daemon* (rpd) is interrupted, service will be impacted.

Command Line Interface (CLI)

After working with an M-series Juniper, the thing you will become most accustomed to by far is the CLI. It is the means with which the user most often interfaces with and queries the router. A great deal of planning and engineering has gone into the CLI to make it intuitive and user friendly. This allows the end user to focus on gathering information and performing job duties rather than muddling with syntax and fighting every command phrase.

As was mentioned before, JUNOS runs a number of independent processes. Of primary concern to the CLI is the *management daemon* (mgd). You can think of the mgd as the process through which the command line hands its queries and requests to other processes. It is more complex than that, but for the purpose of the JNCIA those differences are academic. When a user accesses the router, a CLI process is started for that user and the mgd spawns a child management daemon process to support that user. Therefore, each individual logged into the router has a separate CLI and mgd-child process supporting them.

The CLI can run in two modes: *operational* and *configuration*. After first logging into an M-series, you will be placed in the operational mode. From this main prompt, troubleshooting, diagnostics, and information gathering take place. The configuration mode is a special, restricted set of commands that can be used to modify the router configuration file. Both modes run in the same manner; they use the same control keys, a similar structure, and identical interface. The only difference is the sub-set of commands available in each mode. Let us consider the things common to both modes.

The command hierarchy is broken into a logical tree, beginning at a general level and narrowing to a specific focus. When entering commands into the CLI, EMACS key associations can be used to manipulate the cursor (for some of the more common key-bindings, see Figure 3.1).

Moving the Cursor	
Control-b	Back-up one character
Alt-b	Back-up one word
Control-f	Move forward one character
Alt-f	Move forward one word
Control-a	Move to the beginning of the line
Control-e	Move to the end of the line
Deleting Characters	
Control-h	Delete the character before the cursor
Control-d	Delete the next character
Control-k	Delete all characters from the cursor to the end of the line
Control-x	Delete all characters on the command line
Scrolling through the command history	
Control-p	Cycle backward through the recent command history
Control-n	Cycle forward through the recent command history
Control-r	Search through the command history for a matching string

Figure 3.1 EMACS Editor Keys

JUNOS stores a history of the last commands entered. One can cycle through the history using the control-p and control-n combinations. This is helpful if a number of commands are being cycled through, or if a similar command is issued a number of times requiring only a small change.

In addition to the mundane use of separating commands, each time the space bar is pressed the CLI attempts to parse what is on the command line. Thus, partially typing a command followed by a space will attempt to complete the name of the command automatically. If there is more than one possible completion, pressing the space bar multiple times will echo back all possible endings. For example, typing in “sho” and a space, will allow the CLI to auto-complete “show”. This ensures that a router administrator doesn’t get to the end of a long and complex command string before the CLI errors out. As soon as a non-acceptable command is entered, JUNOS “complains” that it is not a valid input. Erroneous input is underscored with a carat (^) at the first point JUNOS is unable to complete the command. This is, of course, completely a matter of syntax. JUNOS will not complain about poor configuration design if the command structure is correct. So, typing ‘sho osp int’ will echo back ‘show ospf interface’ and then display the command output. But mistyping ‘show opsf’ will return a syntax error because ‘opsf’ is not a recognized command.

```
jncia@my.router> show opsf
                      ^
syntax error, expecting <command>.
```

At any point within monitoring or configuring a router, the “?” key can be used to display all commands available at that level. The “?” can either be used in at the end of a word to see valid completions, or in the place of a word to see all possible selections.

```
jncia@my.router> show os?
Possible completions:
  ospf           Show information about OSPF

jncia@my.router> show ospf?
Possible completions:
  database       Show OSPF link-state database
  interface      Show OSPF interface status
  io-statistics  Show OSPF I/O statistics
  log            Show OSPF SPF log
  neighbor       Show OSPF neighbor status
  route          Show the OSPF routing table
  statistics     Show OSPF statistics
```

Additionally, help files are stored on the router’s hard drive and can be accessed at any time using the commands:

`Help topic` *`topic`* (general information about this subject)

`Help reference` *`topic`* (more detailed information and configuration guidelines).

Operational Mode

As was noted previously, operational mode is where the user is first placed after logging in. This familiar prompt contains the username logged in at the device name followed by a chevron, sometimes referred to as a 'greater than' (>):

```
jncia@my.router>
```

Commands available from the main level of operational mode are:

```
jncia@my.router>?
Possible completions:
  clear       Clear information in the system
  configure   Software configuration information
  file        Perform file operations
  help        Provide help information
  monitor     Real-time debugging
  mtrace      Multicast trace source to receiver
  ping        Ping a remote target
  quit        Exit the management session
  request     Make system-level requests
  restart     Restart a software process
  set         Set CLI properties, date, time,
              display
  show        Show information about the system
  ssh         Open a secure shell to another host
  start       Start a software process
  telnet      Telnet to another host
  test        Diagnostic debugging commands
  traceroute  Trace the route to a remote host
```

The range of 'show' commands are by far the most useful when troubleshooting a network event. They are non-intrusive, informative commands about the protocols, hardware, and processes running on the router.

What follows is a partial list of the more useful commands within JUNOS. For the protocol dependent ones, more detailed descriptions will be given in their respective chapters.

show system uptime	Displays the amount of time the system has been running since the last reload.
show chassis hardware	Reports a list of the installed hardware components and serial numbers.
show chassis fpc	Reports the status of the Flexible PIC Concentrators
show chassis sfm	Reports the status of the Switch Fabric Modules
show log *logfile*	Displays the contents of a configured traceoptions logfile.
show version	Shows the current code revision running on the router.
show configuration	Displays the currently running configuration.
show interface terse	Gives a complete listing of all operating interfaces.
show interface description	Reports the configured name for all interfaces.
show bgp summary	Lists all BGP neighbors and session information.
show bgp neighbor <*neighbor*>	Displays detailed information about a specific BGP peer.
show ospf neighbors	Shows all ospf neighbors that are currently configured.
show ospf interface	Lists all currently active OSPF interfaces.
show isis interface	Lists all currently active IS-IS interfaces.
show route x.x.x.x	Displays current routing information for a specific prefix.
show policy <*policy-statement*>	Lists the policy configured with the given name.
monitor interface traffic	Displays real-time throughput statistics for active interfaces
monitor interface *interface*	Reports real-time traffic, error, and physical layer statistics
monitor interface traffic	Real-time traffic monitoring of all logical interfaces
monitor start <*logfile*>	Logs information to CLI as it occurs. Useful for debugging.
monitor stop <*logfile*>	Ceases the display of *logfile* information to the screen.
monitor stop all	Ceases the display of all logfile information to the screen.

(Note that the ‘clear’ commands will reset all possible targets if one is not specified)

clear interface statistics <*interface-name*>	Resets the traffic and error counters
clear bgp neighbor <*neighbor*>	Resets BGP adjacency
clear ospf neighbor <*neighbor*>	Resets OSPF adjacency
clear rsvp session <*name*>	Resets RSVP adjacency
clear ldp neighbor <*neighbor*>	Resets LDP adjacency
clear mpls lsp <*lsp-name*>	Resets MPLS label switch path.

request support information	Displays verbose technical information used by Juniper to assist with troubleshooting router issues.
request system snapshot	Creates back-up files used to restore a router in event of failure. Useful before conducting maintenance.
request system halt	Gracefully exits active system processes.
request system reboot	Reloads the router.

Configuration Mode

The active configuration (plus several backup configurations, as discussed below) is stored on the routing engine’s compact flash drive. To configure routing and hardware parameters within JUNOS, you enter “edit” or “configuration” mode. The configuration level, like the operational prompt, is hierarchically arranged from less specific to more specific entries. Options available from the main level of edit mode are:

```
[edit]
jncia@my.router#?
Possible completions:
  <[Enter]>    Execute this command
  activate     Remove the inactive tag in a statement
  annotate     Annotate the statement with a comment
  commit       Commit current set of changes
  copy         Copy a statement
  deactivate   Add the inactive tag to a statement
  delete       Delete a data element
  edit         Edit a sub-element
  exit         Exit from this level
  help         Provide help information
  insert       Insert a new ordered data element
  load         Load configuration from an ASCII file
  quit         Quit from this level
  rename       Rename a statement
  rollback     Roll back database to previous version
  run          Run an operational-mode command
  save         Save configuration to an ASCII file
  set          Set a parameter
  show         Show a parameter
  status       Display users currently editing
  top          Exit to top level of configuration
  up           Exit one level of configuration
  update       Update private database
```

As can be seen, the main level does not directly lead to sub-portions of the configuration. This level contains the operators that will add, change, or delete portions of the configuration. Actual protocol, interface, and policy options need to be accessed with their specific statements. Those specific layers of the file can be reached with ‘edit’, ‘insert’, ‘de/activate’ and ‘delete’. The levels of configuration reachable with operators include:

```
jncia@my.router# edit?
Possible completions:
  access              Network access configuration
  accounting-options  Accounting data configuration
  chassis             Chassis configuration
  class-of-service    Class-of-service configuration
  firewall            Define a firewall configuration
  forwarding-options  Control packet sampling
  groups              Configuration groups
  interfaces          Interface configuration
  policy-options      Routing policy option configure
  protocols           Routing protocol configuration
  routing-instances   Routing instance configuration
  routing-options     Protocol-independent options
  security            Security configuration
  snmp                Simple Network Management
                      Protocol
  system              System parameters
```

The branches of the hierarchy to pay most attention to for the JNCIA are the firewall, interface, policy-options, protocols, and routing-options levels. This is by no means a complete list of what is possible with the robust and powerful configuration options. Rather, it is a partial set of what the JNCIA will focus upon. For a complete and detailed discussion of configuration options it is best to consult the Juniper Networks user guides or www.juniper.net.

The configuration highlights that should be studied for firewall and policy-options will be covered in the Policy chapter. Likewise, the study points for protocol configuration will be dealt with in their specific chapters. The remaining topics of interface and routing-options, will be dealt with later in this chapter.

[edit firewall] – This statement allows for the creation of packet filters and options for the firewall policy. This is covered in the *Policy* chapter.

[edit interface] – This statement allows access to configure physical and logical aspects of the interfaces, including addressing, sub-interface units, and encapsulation.

[edit policy-options] – Protocol routing policies, as-path expressions, and prefix-lists are accessed with this statement. See the *Policy* chapter.

[edit protocols] – The various routing protocols are configured at this level. Thisincludes: RIP, BGP, OSPF, IS-IS, LDP, RSVP, MPLS, DVMRP, IGMP, MSDP, and PIM. See the protocol specific chapters as well as the *Multicast* and *MPLS* sections.

[edit routing-options] – The options for configuring static routes, the Router ID (RID), and the BGP autonomous system number (ASN or AS) are accessed with this statement.

Table 3.2 Basic Configuration Statements

For example, routing protocols are configured separately, but they are all covered under the logical group ‘protocols’. To configure BGP, one must enter edit mode and then go to the protocols hierarchy level. From `[edit protocols]`, OSPF, BGP, RIP, and IS-IS may all be accessed, but they are all logically different groups under ‘protocols’.

From the operational CLI, the “show configuration” command will display the configuration in its hierarchical logical tree. Simply typing “show” in configuration mode will give the same output. Once under any sub-level of “edit” configuration hierarchy, such as `[edit protocols bgp]`, typing “show” will not list the entire configuration but rather will list the configuration for that level of the configuration only. For example:

```
[edit protocols bgp]
jncia@my.router# show
    group internal {
        type internal;
        local-address 192.168.1.1;
        neighbor 192.168.1.100;
    }
}
```

Curly braces separate information with the configuration, much akin to C programming. An open brace, also called a left brace ({), indicates the beginning of a logical grouping of configuration commands. In order to be syntactically correct, the closing brace (}) must follow and end the last part of the group. More simply, every { must be followed by a } at some point.

Below is a look at this with an expanded view of how ‘show configuration protocols’ might appear for a very simple configuration. Don’t concentrate on the portions of the configuration that might be unfamiliar, but rather try to locate braces and the logical grouping of components.

```
jncia@my.router>: show configuration protocols

protocols{                                This brace begins ‘protocols’

bgp {                                     This brace begins ‘bgp’
   group internal {                       Begins/Opens group ‘internal’
      type internal;
      local-address 192.168.1.1;
      neighbor 192.168.1.100;
   }                                      Closes/Ends group ‘internal’
}                                         Closes ‘bgp’
ospf {                                    Begins ‘ospf’
   area 0.0.0.0 {                         Opens ‘area 0.0.0.0’
      authentication-type md5;
      interface so-3/0/0.0 {              Opens ‘interface so-3/0/0.0’
         metric 20;
         authentication-key
      }                                   Closes ‘interface’
   }                                      Closes ‘area 0.0.0.0’
   area 0.0.0.3 {                         Opens ‘area 3’
      authentication-type md5;
      interface so-4/0/0.0 {              Opens ‘interface so-4/0/0’
         metric 20;
         authentication-key
      }                                   Closes ‘interface’
   }                                      Closes ‘area 3’
 }                                        Closes ‘ospf’
}                                         Closes ‘protocols’
```

Figure 3.3 ‘Show Configuration Protocols’ Sample

To modify or add portions to the configuration from the command line the ‘set’ syntax is used. If we were to configure the router-id and AS number by means of the CLI, we can utilize ‘set’ commands. Both of these options are defined under routing-options. To correctly configure them, one would:

Enter edit mode.

```
jncia@my.router> configure
Entering configuration mode

[edit]
jncia@my.router#
```

Notice the prompt is now below a header that indicates our relative position in [edit] and has changed to a hash (#). The CLI has now entered configuration mode. Because the options that will be changed are at the routing options level, we will make the changes there:

```
[edit]
jncia@my.router# edit routing-options

[edit routing-options]
jncia@my.router#
```

Notice how we are now at the `[routing-options]` level of configuration and how that is now indicated in the prompt. From this level we can more easily access all of the options below this tier, but none of the ones above it. Adding the Router ID is now a matter of using *set*.

```
[edit routing-options]
jncia@my.router# set router-id 10.10.100.101
```

'Set' can be used at any level along the tree as long as a valid path to that option exists from the present location. Remember since we are at `[routing-options]` we can access and modify the information at this level and below, but cannot access anything that is above it. For instance, if we need to change the router hostname, we will not be able to access it from this level.

```
[edit routing-options]
jncia@my.router# set hostname
                     ^
syntax error.
```

The hostname option is under the `[system]` portion of the configuration tree, and thus is not configurable from here. Indeed, JUNOS will not allow you to proceed typing a hostname at this point. Attempting to enter the space after hostname will likewise echo back a syntax error as the software cannot find a valid completion for that command at this level.

It is important to note that configuration commands within JUNOS are somewhat different than Cisco IOS in that the exact command you type does not appear in the configuration file.

Examine this example of a basic BGP configuration:

```
[edit protocols bgp]
jncia@my.router# show

     group ebgp-peer {
          peer-as 123
          type external
          neighbor 10.0.0.1
     }
```

You see the familiar hierarchical tree format. However, to set these parameters you could simply type one line:

```
[edit]
jncia@my.router# set protocols bgp group ebgp-peer
peer-as 123 type external neighbor 10.0.0.1
```

Or, you could enter these commands one at a time:

```
[edit]
jncia@my.router# edit protocols bgp group ebgp-peer

[edit protocols bgp group ebgp-peer]
jncia@my.router# set peer-as 123

[edit protocols bgp group ebgp-peer]
jncia@my.router# set type external

[edit protocols bgp group ebgp-peer]
jncia@my.router# set neighbor 10.0.0.1
```

Notice how neither of these resemble what is displayed when typing ‘`show`’. The command statements have been parsed and placed in the correct logical grouping with the correct braces. JUNOS takes care of all that. Note as well that the entire subsection could be entered with four command statements, or with one slightly longer, more complex `set`. Finally, take note of the fact that in the second example we move down the command tree several levels at once, four levels to be exact: [protocols], [bgp], [group] and the group-name [ebgp-peer]. This could have been accomplished in four separate steps as well. Just as the first example configured multiple options at once, the second moves down several levels at once.

We have seen how entering `[edit <sub-section>]` allows us to progress down the tree. There are three ways to travel back up the command hierarchy as well: `exit, up,` and `top`.

Exit – This returns the user to the last level that was active. If the user had gone to a [edit protocols bgp group ebgp-peer] with one command, exit will return the CLI to the top level. If the user had gone to [group ebgp-peer] from [protocols bgp], exit will return to [protocols bgp]. Exiting from the main configuration CLI will return the user to operational mode.

```
     [edit]
     jncia@my.router# edit protocols bgp group
ebgp-peer

     [edit protocols bgp group ebgp-peer]
     jncia@my.router# exit

     [edit]
     jncia@my.router# edit protocols bgp

     [edit protocols bgp]
     jncia@my.router# edit group ebgp-peer

     [edit protocols bgp group ebgp-peer]
     jncia@my.router# exit

     [edit protocols bgp]
     jncia@my.router# exit

     [edit]
     jncia@my.router# exit
     Exiting configuration mode

     jncia@my.router>
```

Up – This command moves to the level immediately above the present tier by a specified number of steps. If the user had gone to specific level four lower than the general top level of edit, up will move the CLI to a level three lower than general edit. Entering 'up 4' will have the same effect as exit. The default is one level.

```
     [edit]
     jncia@my.router# edit protocols bgp group ebgp-peer

     [edit protocols bgp group ebgp-peer]
     jncia@my.router# up (Note that this moves up both
[group] and [group name], as they are logically one level)

     [edit protocols bgp]
     jncia@my.router# up

     [edit protocols]
     jncia@my.router#
```

Top – This command moves to the root, or main, edit prompt regardless of current level or how many steps were taken to get there.

```
     [edit protocols bgp]
     jncia@my.router# top

     [edit]
     jncia@my.router#
```

Candidate Configuration and Commit

Two extremely useful points regarding system configuration are the idea of the candidate configurations and the `'commit'` statement.

Once in the configuration mode `[edit]`, the active configuration can be modified in any way. As soon as you enter configuration mode, the aspects you are altering become known as the *candidate configuration*. The candidate configuration does not replace the running configuration until the 'commit' command is entered. If there is a syntax error upon committing the candidate, the process will report an error and continue running from the pre-commit configuration. Note as well that if the configuration is modified and the user exits from edit mode without committing the changes, the router will still be running the original configuration but will have a modified candidate buffered in edit. If someone later re-enters edit mode to make different changes, they may well end up unknowingly committing changes someone else had made previously.

Loading files

There are also a number of ways to import sections of a configuration from ASCII files with the `load` command. This takes the requested file and places it in the candidate configuration. It is important to note that the ASCII file must have the configuration in the correct hierarchical format. This means that JUNOS is expecting commands that are in the correct tree syntax, complete with accurate brace placements around the correct groupings ({,}). A file filled with '`set`' command statements will not work with '`load'`. There are three options when loading a file:

```
[edit]
load [merge|override|replace] [terminal|filename]
```

Load merge – This command statement adds the contents of the file to the current configuration and places the combined file into the candidate. If there are any conflicts between the loaded file and the current one, the loaded parameters over-write them.

Load override – This command disregards the existing configuration. The requested file, and the requested file alone, becomes the candidate. For this reason it is imperative that the loaded file contain a **complete** configuration.

Load replace – This final possibility copies the current configuration into the candidate and attempts to match and substitute the contents of the loaded file. The loaded file must contain the `replace:` tag prior to the configuration options changed to work correctly.

One further option that may be requested is to manually enter the data rather than load a pre-typed file. By specifying '`terminal`' as the source rather than a file name, the configuration CLI allows a user to input data in a field and terminate input with control-d when finished. This data field becomes the source file for the candidate load. Again, this data must be in the correct hierarchical format, complete with braces. For this reason, this option is not really meant for a user to manually type in the data; instead it is a great way to quickly cut-and-paste in a desired pre-assembled snippet. Manually configuring the router is always best done with `set` commands.

Rollback

There have been a number of notes mentioned about how JUNOS keeps backup copies of previous configurations on hand for quick access. As was mentioned previously, any configuration changes made to the router do no take effect until a commit is performed. But what happens if there is an unforeseen problem after the commit is made? With certain types of network gear, there might be no choice but to manually undue each of the configured changes by explicitly deleting each change that was made. In extreme cases, an entire backup configuration might have to be loaded. JUNOS eases this not altogether uncommon problem of configuration mistakes and allows for easy reversal. By keeping the latest configuration handy in flash memory, if a problem is encountered the technician can immediately take advantage of the *rollback* command to reload the pre-changed file. In a sense, it is a shortcut `load override` with a locally stored file.

```
[edit]
jncia@my.router# rollback ?
Possible completions:
  <[Enter]>      Execute this command
  <number>       Numeric argument
  0              2003-03-11 09:33:44 UTC by john via cli
  1              2003-02-13 09:32:52 UTC by jeff via cli
  2              2003-02-09 18:29:02 UTC by tyler via cli
  3              2003-01-20 21:18:23 UTC by root via cli
  |              Pipe through a command
```

The rollback will replace the modified config file with the back-up. This is the equivalent of loading any locally stored file and must still be committed for the router to take it. Because the router keeps the 9 most recent modifications, a complex, multi-step maintenance gone wrong can still be somewhat painlessly backed out. This should not, however, be considered an excuse for not having adequate remote back-ups of configuration files. Accidents and system crashes do occur, as do maintenances with a dozen modifications.

We can see from the previous example that the configuration was modified four times. The actively running configuration file is numbered 0. We can also see a timestamp for when changes were made. Upon entering:

```
[edit]
jncia@my.router# rollback 1
```

File number 1 is loaded as the candidate configuration. It will not take effect until a `‘commit’` is issued. Note that rollback commands must be given from the top level of `[edit]`.

If we were instead to edit and commit an additional change, each number would increment by 1. There would be a new active file 0 written with the current timestamp, and there would then be 5 files total.

Traceoptions

One of the unique and extremely helpful features of the JUNOS software is the ability to setup and monitor detailed logging information. This logging information is known in JUNOS as *traceoptions*. They are extremely versatile and may be applied at practically any and all levels of the configuration. The log information is stored, by default, on the router hard drive. The default logfile is named *messages* and it contains information about interface and protocol flaps as well as general error output. Additional logs and more specific traceoptions, called flags, can be configured. Logs can be viewed by issuing the “show log *logfile-name”* command at the operational CLI. Additionally, “show log?” will display the available files that have been configured for traceoptions. Another useful command is “monitor start *logfile-name*”, which continually prints new output to the file as it is posted to the log. This is extremely useful for real-time debugging of routing problems. The display of this information can be stopped by “monitor stop *logfile-name*” or “monitor stop all”.

Introduction to Routing

Routing is the process by which data is delivered to and through the correct nodes in a network. The rules followed by particular nodes that make routing possible are called *protocols*. A *routing* protocol is one which determines paths and delivery, or routes, of data packets. Examples of routing protocols include OSPF, BGP, IS-IS, RIP, and the Cisco proprietary E/IGRP. A *routed* protocol is one which carries the data, such as IP, IPX, Appletalk or the like. Juniper routers are optimized for routed IP, as indeed is most of the Internet, and as such the Juniper certifications are based upon IP only. Similarly, Juniper routers utilize open standards routing protocols. Consequently the JNCIA will exclusively focus upon RIP, OSPF, IS-IS, BGP and multicast protocols.

The JUNOS software architecture maintains routing information in two related databases: the *Routing Table* and the *Forwarding Table*. The Routing Table contains all routing information learned by all routing protocols running on the Juniper. The Forwarding Table contains only the routes that are actually used to forward packets to their destinations. In a sense, the Routing Table holds all possible routing information for all destinations known to the router. Meanwhile, the Forwarding Table holds only the best path per destination, called the *active route*. By only holding the active route, the Forwarding table is minimized in terms of size, while still being able to reach every known location.

By default, JUNOS software maintains multiple routing tables. The following are the important tables to remember for the exam:

- inet.0 - Default unicast table
- inet.1 - Default multicast table
- inet.2 - Multicast RPF checks
- inet.3 - MPLS path information
- mpls.0 – MPLS label-swapping table

(*inet* is an abbreviation for Internet)

The mpls.0 table lists the next hop router for each label switch path, allowing transit routers to forward packets along the LSP. See the *MPLS* section for more details.

More detailed information regarding the process prefixes travel through before becoming installed into the forwarding table will be covered more in-depth in the *Hardware* section.

There are a number of different routing protocols with which a router might learn how to get to a particular address. Because of this, the device needs a system to determine which route it will prefer. Each prefix put into the routing table of a Juniper router is listed along with the protocol from which it is learned. While the JUNOS software might have many routes to a particular host, only the active route is installed into the forwarding table and used to route packets. Each protocol is assigned a *preference*, a number from 0-255, with lower numbers being more preferable. Generally, a better preference indicates a more reliable protocol and hence a more desirable path. The table below illustrates how the protocol preferences are broken down.

How route is learned	Default Preference
Directly Connected	0
Static	5
MPLS	7
OSPF Internal	10
IS-IS Level 1	15
IS-IS Level 2	18
Redirects	30
RIP	100
Point to Point	110
Aggregate	130
OSPF External	150
BGP	170

Table 3.4 Route Preference

So if a router knows of two paths to a particular destination, one through RIP and another via OSPF, it will choose OSPF with a preference of 10. It may seem a bit counter-intuitive to think lower preference is better, but this is one of the few cases where this is true. Keep in mind that these preferences can be modified from their default value through configuring *policy* (to be covered later in this book).

If a router has more than one equal-cost path to the exact same destination, it will randomly install one of those routes into the forwarding table. In order to install multiple routes, policy must be applied to load balance traffic between the paths.

Static vs Dynamic Routes

Routing protocols such as OSPF are examples of how a router learns *dynamic routes.* Dynamic routing utilizes information exchanged by protocols between neighboring routers to monitor and modify route information as the network topology changes. Routes may be manually configured by an administrator. Such routes do not change unless the configuration is updated and do not reflect the overhead 'work' of a dynamic routing protocol. Such routes are called *static routes*. The configuration for static routes is relatively straight-forward and simple, but it does not provide a robust system of routing or the advantages of automated redundancy common to dynamic protocols.

Once configured, static routes remain active and do not respond to topology changes. The exception to this is if the change is local to the router, such as if the static route is associated to an interface or next-hop that goes away. Static routes are installed in inet.0.

Configuring Static Routes

Static routes do not have a protocol per se, and are therefore configured under the [routing-options static] level of edit mode. There must be a valid next-hop address for the static address to work. A null value is considered valid for this. The syntax is:

```
[edit routing-options static]
jncia@my.router# set route destination-address/subnet-
mask [discard|reject|next-hop <next-hop-
address>|<interface-name>]

[edit]
jncia@my.router# show routing-options

routing-options{
      static {
            route 192.168.100.0/24 next-hop 10.1.134.2;
            route 192.168.200.0/24 next-hop so-1/0/0.0;
            route 192.168.220.0/24 discard;
            }
}
```

If an IP address or interface is not given for a next-hop, one of the null values must be used. When a null route is configured, packets matching that route are dropped. The difference between the `reject` and `discard` null value is in how they respond to dropping the packet. A `reject` null route will return an ICMP "prohibited" message. A static route with a `discard` null value will silently drop the packet, returning no message.

There is a fourth option, `receive`, for static routes. This indicates that packets matching the static are to go to the RE. Static routes with the `receive` option are not covered on the JNCIA.

Each of the dynamic protocols on the JNCIA exam will be covered in a dedicated chapter later in this book. The advantages and challenges of each will be detailed at that time. At this point, suffice it to say, the additional configuration complexity and overhead are generally accepted because of the benefits of less administration and dynamic fault tolerance. The router uses static routes when it does not have a route with a lower (better) preference (see the above), when it cannot determine the route to a specific destination, or when forwarding un-routable packets.

It should also be noted that the individual routing protocols can be manually configured to have a different preference, thus allowing a different route to be selected. An example of this would be someone configuring a static route of last resort and stipulating a preference of 200. This way, any other identical route coming from a dynamic protocol will be selected first.

Interface and Routing-Options Configuration

As was mentioned before, protocol and policy relevant configuration will be covered in their specific chapters. The configuration areas that remain and should be given consideration for the examination will be covered here.

Interface Configuration

One of the two main levels of configuration that need attention is:

[edit interfaces] – Access to configure physical and logical aspects of the interfaces

Configuration options at the interface level of edit mode allow the user to define interfaces addresses, sub-interfaces, encapsulations, and the routing family with which it will forward packets. Juniper routers have two types of interfaces, *permanent* and *transient*. Any interface installed as a PIC is a transient interface. This includes all the mixes of SONET, ATM, Ethernet, and serial ports. Permanent interfaces are automatically detected by JUNOS and begin with the identifier *fxp*. The management Ethernet interface for out of band connectivity is `fxp0`. The internal Ethernet interface that allows communication between the RE and the PFE is `fxp1`.

When installing transient interfaces, JUNOS automatically recognizes the hardware and provides a physical tag for it. The physical interface tag is given in the format of `<type>-<fpc>/<pic>/<port>` where:

- Type = The media type (a complete list is in the *Appendix*)
    a) so = SONET
    b) at = ATM
    c) fe = Fast Ethernet
    d) ge = Gigabit Ethernet
    e) lo = Loopback
    f) t1 = T1 interface
    g) t3 = T3 interface
- FPC = The FPC slot in which the PIC resides
- PIC = The PIC location on the FPC
- Port = The PIC port number

When reading the physical interface tags, it is important to remember that numbering begins with zero. For example, `so-0/1/2` describes the physical SONET interface that is located on the first FPC slot (numbered 0), PIC slot 1, and port 2 on that PIC. This physical tag is used for identifying the particular interface a user wishes to view or configure and is not the same as the interface description, which may be set with the `description` option under `[edit interfaces]`. It is user defined and may be any string. However, JUNOS requires any string that includes spaces to be enclosed in double quotes (“description”).

```
[edit interfaces]
jncia@my.router# set so-4/0/0 description “OC3 to
Chicago”

[edit interfaces]
jncia@my.router# show

so-4/0/0 {
        description "OC3 to Chicago";
}
```

The physical interface requires an encapsulation type to successfully establish a layer-2 link with the remote interface. Encapsulation types vary by physical interface, but they are all configured under the interfaces level of edit mode. Enabling Point-to-Point protocol on our sonet link looks like:

```
[edit interfaces]
jncia@my.router# set so-4/0/0 encapsulation ppp

[edit interfaces]
jncia@my.router# show
    so-4/0/0 {
        description "OC3 to Chicago";
        encapsulation ppp;
}
```

Each physical interface may have one or more logical, or virtual, interfaces mapped to it. Multiple logical sub-interfaces can be useful in ATM, Frame Relay, and Ethernet networks when creating virtual circuits or VLANS on a single physical port.

These sub-interfaces are configured under `[edit interfaces]` level with the `unit` command statement.

```
[edit interfaces]
jncia@my.router# set so-4/0/0 unit 0

    so-4/0/0 {
        description "OC3 to Chicago";
        encapsulation ppp;
        unit 0
}
```

Each logical interface descriptor can have one or more *family* descriptors to define the protocol family that is associated with and allowed to run over the logical interface. Families referred to for the JNCIA are listed below. A complete list is included in the *Appendix*.

- Internet Protocol, version 4 (IPv4)
- International Organization for Standardization (ISO)
- Multiprotocol Label Switching (MPLS)

The Internet family, commonly referred to as 'family inet', is used when considering IP protocols such as OSPF, BGP, and RIP. 'Family iso' is used in conjunction with IS-IS, while 'family mpls' is, of course, used by MPLS forwarding. Multiple families may be configured on a single interface. Simply put, this means that the interface must be configured for the types of protocol families that will be used upon it. An interface must have a family type enabled before it will allow network traffic to utilize the link.

```
[edit interfaces so-4/0/0 unit 0]
jncia@my.router# set family inet

[edit interfaces so-4/0/0 unit 0]
jncia@my.router# set family mpls

    so-4/0/0 {
        description "OC3 to Chicago";
        encapsulation ppp;
        unit 0 {
             family inet;
             family mpls;
        }
    }
```

Additionally, the address for a particular family is designated beneath the family level of configuration.

```
[edit interfaces so-4/0/0 unit 0]
jncia@my.router# set family inet address
192.168.10.1/30

    so-4/0/0 {
        description "OC3 to Chicago";
        encapsulation ppp;
        unit 0 {
             family inet {
                 address 192.168.10.1/30;
             }
             family mpls;
        }
    }
```

Configuring Routing-Options

The second area of configuration that does not fit under a specific protocol or other area is that of routing-options. There are three areas important to the JNCIA at this configuration level of which you should be aware.

```
[edit routing-options]
jncia@my.router# set?
Possible completions:
      autonomous-system  Autonomous system number
      router-id          Router identifier
      static             Static routes
```

(These are not all the possible completions, only the ones that may concern the examination.)

We have already seen the concept of static routes and the methods for configuring them previously in this chapter.

```
[edit]
jncia@my.router# show routing-options

routing-options{
      static {
            route 192.168.100.0/24 next-hop 10.1.134.2;
            route 192.168.200.0/24 next-hop so-1/0/0.0;
            route 192.168.220.0/24 discard;
            }
}
```

The Autonomous System Number (ASN) is necessary for BGP to function properly. However, since it actually defines the administrative group to which the router belongs, it is configured below routing options:

```
[edit routing-options]
jncia@my.router# set autonomous-system as_number
```

Where `as_number` is the desired ASN 1 through 65535 inclusive.

Likewise, the Router ID (RID) is a crucial component used by a number of processes. It is defined with the command statement below:

```
[edit routing-options]
jncia@my.router# set router-id address
```

Key Points

- JUNOS runs on a UNIX kernel.
- Software processes run in separate memory space for increased stability.
- The main JUNOS image resides on compact flash.
- A back-up image is kept on the hard-drive.
- The standard boot sequence for JUNOS:
    - PCMCIA flash
    - Compact flash
    - Hard-drive
    - Network
- The CLI has an operational and configuration mode.
- Static routes are manually configured and go into inet.0.
- Static routes must have a valid next-hop or null value configured.
- Dynamic routes are learned via routing protocols.
- JUNOS uses administrative preference to choose between multiple routes. Lower preference is better.

Directly Connected	0
Static	5
MPLS	7
OSPF Internal	10
IS-IS Level 1	15
IS-IS Level 2	18
Redirects	30
RIP	100
Point to Point	110
Aggregate	130
OSPF External	150
BGP	170

- Interface configuration controls:
    - Sub-interfaces
    - Protocol families
    - Addressing
    - Encapsulation
- Routing-option configuration controls:
    - Static routes
    - Autonomous System Number (ASN)
    - Router ID (RID)

Chapter Four

RIP

Chapter 4: RIP

Overview

In this chapter, we will discuss Routing Information Protocol (RIP). We will cover the differences between RIP version 1 and version 2, functionality, limitations, and configuration on a Juniper router.

RIP is arguably the easiest dynamic routing protocol to configure and run. Many of the concepts that follow will also be present in the more robust, complex protocols that appear later on in the book. By the end of this chapter, you should be able to understand and apply the following concepts:

- ✓ Understand why administrators implement dynamic routing
- ✓ RIP best path selection.
- ✓ Establishing and maintaining RIP neighbors.
- ✓ Route updates in RIP.
- ✓ Routing loop prevention within RIP.
- ✓ Configuration of RIP in JUNOS.

Introduction

You have already learned how the Juniper router organizes destination prefixes in the routing table and uses that to establish the forwarding table. Additionally, you have seen how to manually configure a next hop, called *static routing*. We now move on to cover the idea of *dynamic routing protocols*. Routing protocols are just that, rules by which two systems can exchange information on how to route packets. Unlike static routes, which must be manually updated, dynamic protocols can learn about topology changes, add new routes, and optimize routing without outside intervention. This translates into a more scalable network with less administrative overhead. There are a number of different routing protocols, each one offering its own advantages and challenges.

Dynamic routing protocols are generally grouped into two types: *distance vector* and *link-state*. We will discuss the concepts behind link-state protocols in the IS-IS and OSPF sections. RIP is a distance vector protocol. Distance vector protocols use *hop count* to determine the distance to get to a network, where each router that the traffic passes through is considered a hop. In this way, routers can determine which path will cross the least number of nodes. This is assumed to be the quickest/best path. RIP routers advertise their directly attached networks and the distance (hop count) to the networks they can reach via their RIP neighbors. Each neighboring router that hears advertised routes will then add one to the hop count and advertise them to its neighbors. This is how RIP routers form their view of network topology.

Protocol Fundamentals

Routing Information Protocol (RIP) is a simple distance-vector routing protocol used in fairly small and/or stable network environments. Timers are implemented at set intervals to communicate with its determined neighbors to exchange routing information and updates. RIP uses UDP port 520 for communication between network hosts and the Bellman-Ford algorithm to determine best path selection. Unlike other *Interior Gateway Protocols* (IGPs), RIP uses only a fixed metric (*hop count*) to select a route. The longest network path cannot exceed 15 hops with this routing protocol. A route metric of 16 or more is unreachable. This is what makes RIP best suited for use in smaller, stable networks

Version 1 vs. Version 2

Everything you hear and read about RIP will most likely be referring to version 2. Version 1 is old technology and is not widely used. However, it is important to know the differences between these versions for the JNCIA exam.

Version 1 does not support classless routing. This is to state that the route received by a RIPv1 neighbor is assumed to be class-full. If a version 1 router receives an update for the route 120.1.2.3, it will be automatically assumed to be 120.0.0.0/8 (the Class A equivalent route). This precludes the use of VLSM in RIP version 1.

In version 1, the neighbor’s address is always assumed to be the next-hop address for the destination (not efficient in an Ethernet or other multi-access environment). Version 2 adds a field in route updates for the next-hop address.

Version 1 broadcasts messages. Version 2 uses multicast to the specific address of 224.0.0.9. All RIPv2 routers listen for this address, but do not forward the messages.
Version 2 supports route-tags (information carried from protocols redistributed into RIP), and is backward compatible with Version 1.

Authentication was not introduced until RIP version 2. The RFC for version 2 specifies plain-text passwords, but JUNOS allows plain-text and MD5 encrypted keys.

Communication between RIP hosts

As noted earlier, RIP uses UDP for all communication. This means that the communication of messages is "connectionless and unreliable". That is to state, there is no confirmation when routing updates are sent out to a neighboring host. Because of this fact, timers are padded to allow for the occasional loss of messages. Once an interface is configured for RIP, it will immediately begin transmitting its known and configured networks to all established neighbors. The transmit interval is user configurable, with the default set to occur every 30 seconds.

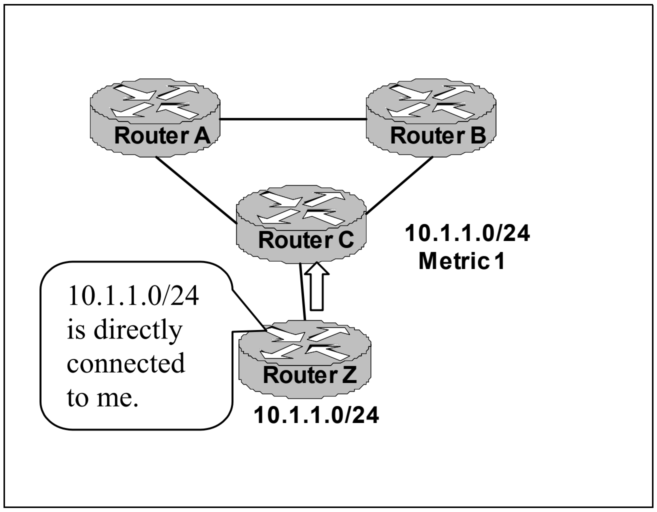


Figure 4.1 RIP Route Update

In Figure 4.1, Router Z is directly connected to network 10.1.1.0/24 and begins to advertise this in its updates with a hop-count of zero. Router C receives this update and increments the metric by one, indicating the destination is one hop away. Checking its routing table and seeing it has no better path, it installs the route and passes the information along in its own route updates that it sends to its neighbors.

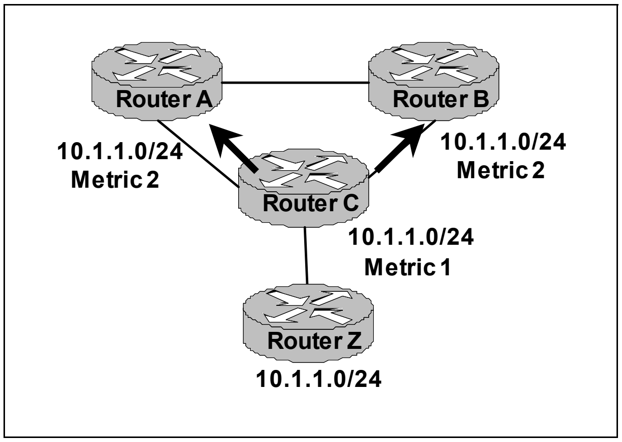


Figure 4.2 RIP Route Propagation

In Figure 4.2, C updates neighbors A and B with the route. They in turn increment the hop count and install the route as well. In the simple topology of 4.2, A and B will update each other with the route to 10.1.1.0/24. However, since the hop count will again be incremented, the metric for that path will then be three. Neither A nor B will choose to install that update as it already has a valid route with a lower metric of 2 by way of going through C.

Neighbors continue to pass RIP route updates hop by hop throughout the network. No neighbor will have a complete view of the topology until it has received all the updates. For this reason, RIP is sometimes called 'routing by rumor', as a router is only aware of what its neighbors are telling it. It is easy to see that this behavior, coupled with the 30-second update timer, could result in very large convergence times. A change propagating over 10 routers would require up to 300 seconds to be fully communicated throughout the network. However the period of the update timer must be balanced with the conservation of network overhead. Because RIP updates include the entire contents of the routing table they can be quite large. Forcing the timers to speed up can easily congest low bandwidth links and spike CPU usage on routers.

Another problem to contend with in RIP, as well as other dynamic protocols, is the controlling the spread of incorrect route information. RIP uses *split horizon with poisoned reverse* to control routing loops within a network. These principals are outlined below:

1) Split Horizon - Omit routes learned from a neighbor when sending updates to that neighbor. This means when Router A advertises networks to Router B, Router B will not send them back to Router A.

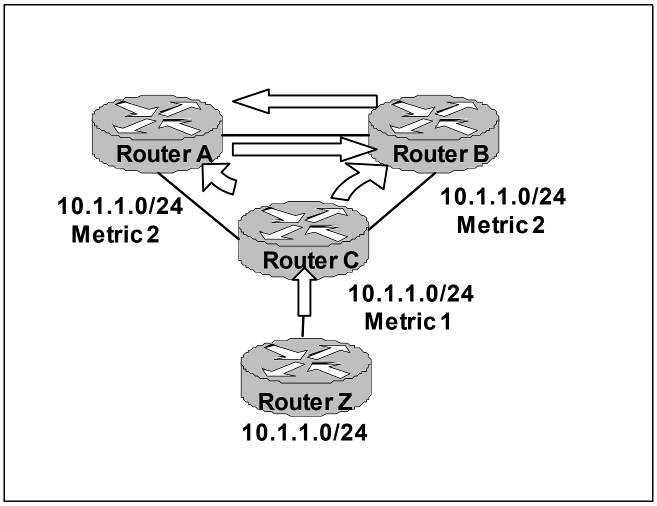


Figure 4.3 RIP Split Horizon

In figure 4.3 the destination 10.1.1.0/24 is directly connected to Z. RIP Router Z advertises its route to 10.1.1.0/24 to Router C, which increments the hop count and gives the route a metric of 1. Router C is running RIP with neighboring routers A and B. It advertises the route to both A and B. Both add 1 to the hop-count bringing the total metric to 2. Split Horizon dictates that Router C does not advertise the route back to Z, since it was received from that router. Similarly, neither A nor B will advertise the route back to Router C because of Split Horizon. Again, is noteworthy to see that Router A and B will announce the destination block to each other. However, because the direct path to Router C has a lower hop-count it wins.

2) Poisoned Reverse - Include such updates, but set their metric to infinity to immediate stop a routing loop. Keep in mind that poisoned reverse has the benefit of more quickly eliminating erroneous routes by marking them as "unreachable", but it does increase the size of routing updates.

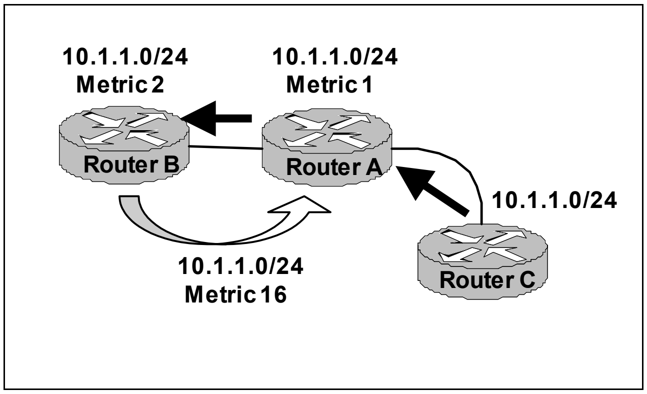


Figure 4.4 RIP Poison Reverse

RIP Packets

RIP messages contain the following fields:

Command—Indicates whether the packet is a *request* or *response* message. Request messages seek information for the router's routing table. Response messages are sent periodically and also when a request message is received from a neighbor. Periodic response messages are called *update messages*. Update messages contain the command and version fields and 25 destinations (by default), each of which includes the destination IP address and the metric to reach that destination.

A request is used to ask for a response containing all or part of a router's routing table. Normally, requests are sent by routers which have just come up and are seeking to fill in their routing tables as quickly as possible. The request is processed entry by entry. If there are no entries, no response is given. There is one special case. If there is exactly one entry in the request, and it has an *address family identifier* (AFI) of zero and a metric of infinity (i.e., 16), then this is a request to send the entire routing table.

Version number—Version of RIP that the originating router is running

Address family identifier (AFI)—Address family used by the originating router (value always “2” for IPv4)

Address—Destination network IP address

Metric—Value of the metric advertised for the address

Mask—Mask associated with the IP address (RIP Version 2 only, value of 0 in RIPv1)

Next hop—IP address of the next-hop router (RIP Version 2 only, value of 0 in RIPv1)

Controlling routing updates

Routes received from routers that are not configured as neighbors are simply ignored. Route updates, when received, can contain up to 25 destinations. If authentication is used, only 24 destinations can be sent in each update message. These updates are transmitted every 30 seconds (by default) by RIP-enabled routers, this is known as the *keepalive*. If four times the keepalive interval is exceeded, the neighbor is considered to be dead.

It is important to recognize that the RIP table remains constant unless one of the following occurs:

1) **Route removal**
    1) The timer expires (default setting 120 seconds) with no updates to reset it.
    2) An update is received with a metric of 16 (thus triggering the route as unreachable). This can also happen via the “poisoned-reverse” process, too.
2) **Route addition**
    1) New network(s) are configured on the router
    2) A new route is received from a valid neighbor with a metric of 15 or less (a reachable route)

Routes received from valid neighbors are compared to existing routes and the following actions are taken:

1) If the advertising neighbor and metric are the same, re-initialize the timeout and wait for the next update
2) If the advertising neighbor is the same but the metric is lower, install the new route and trigger an update
3) If the advertising neighbor is the same but the metric is higher, look closer at the metric
    a. If it is higher, but less than 16, discard the update
    b. If it is 16, start the *route deletion* process for this route

The route deletion process consists of the following steps:

1) Set the new route metric to 16.
2) Trigger an update for this route (even if the timer is not yet due).
3) Start the 'garbage-collection' timer.
    a. Until this timer expires in 120 seconds, send updates for this route with a metric of 16
    b. When this timer expires, delete the route from all tables
    c. If a reachable route arrives during the "garbage-collection" period, the new route will be installed and the timer will be stopped. An update is triggered for this new route.

As an example, in RIP, every gateway that participates in routing sends an update message to all its neighbors once every 30 seconds. Suppose the current route for network N uses Router A. If we don't hear from N for 180 seconds, we can assume that either the gateway has crashed or the network connecting us to it has become unusable. To remove a route from the routing table, a host sends an update to its neighbor with a metric of one higher than the maximum hop count. The receiving host sees this value as 'infinity' and therefore marks the route as unreachable.

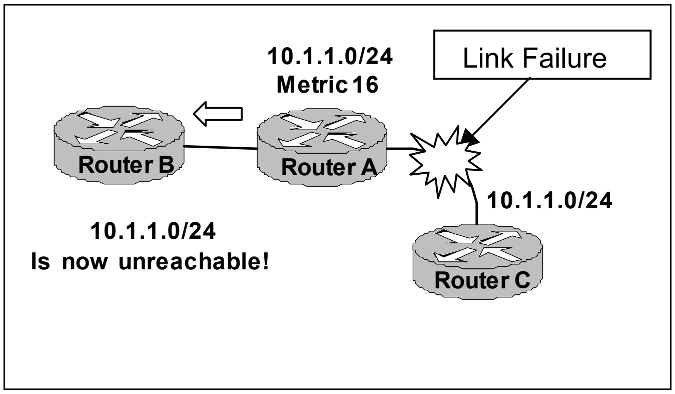


Figure 4.5 Route Removal

Configuration

By default, RIP is disabled on Juniper routers. To enable, enter the following:

```
[edit protocols rip]
jncia@my.router# Set group group-name neighbor
interface
```

All other RIP configuration statements are optional. This minimum configuration defines one group. Include one neighbor statement for each interface on which you want to receive routes. The local router imports all destinations from this neighbor and does not advertise routes (by default). RIP routes received from routers not explicitly configured as neighbors will be ignored. The router can receive both Version 1 and Version 2 update messages.

Once enabled, JUNOS is set to be compatible with both versions by default. JUNOS will send RIPv2 messages, but it will use broadcast addresses so that RIPv1 neighbors will hear the messages. JUNOS will receive both RIPv1 and RIPv2 messages. These parameters can be changed on a global or per-neighbor basis according to the following:

Send Compatibility Parameter Configured via ‘send *X*’

Option	JUNOS term (*X*)	Resulting Action
RIPv1	Version-1	Only RIPv1 messages are sent via broadcast
RIPv2	Version-2	Only RIPv2 messages are sent via multicast
RIPv1 compatible	Broadcast	Only RIPv2 messages are sent, but broadcast addresses are used so RIPv1 neighbors will hear them
Nothing	None	No RIP messages are sent

Receive Compatibility Parameter Configured via ‘receive *Y*’

Option	JUNOS term (*Y*)	Resulting Action
RIPv1 only	Version-1	Only RIPv1 messages are accepted
RIPv2only	Version-2	Only RIPv2 messages are accepted
Both	Both	Accept both RIPv1 and RIPv2
None	None	No RIP messages are accepted

Table 4.6 RIP Send and Receive Parameters

Additionally, the metric (hop-count) of received and advertised routes can be manually adjusted to a specific number through the use of the `metric-in` and `metric-out` configuration statements under the `[edit protocols rip]` hierarchy level.

Let’s look at a typical JUNOS RIP configuration that includes the optional parameters for sending and receiving announcements as well as metric adjustment:

```
[edit protocols rip]
jncia@my.router# show
traceoptions {
  file rip size 1m files 3 world-readable;
  flag policy send;
}
metric-in 3;
```

```
  metric-out 2;
  group rip-neighbor-on-fe0-1-0 {
       metric-out 3;
       neighbor fe-0/1/0.0 {
            send broadcast;
            receive both;
       }
  }
```

This is an example of the multi-tier hierarchy in editing the JUNOS configuration. In the case of RIP, like with other parameters, the more specific instance will nullify the more general. Consequently, `metric-in` and `metric-out` statements applied at the *global* RIP level will be overridden by those configured at the `group` level. Likewise, those at the `group` tier can be overridden at the `neighbor` configuration level.

Policy in RIP

As was noted before, by default JUNOS will accept all RIP routes received from a configured neighbor but will not advertise any known networks on the activated interfaces. Explicit *export* policy must be applied to force routes to be advertised to RIP neighbors as well as *import* policy to filter incoming routes. This practice of building filters with policy to control advertised and received updates is common across the different routing protocols.

To filter routes being imported by the router from RIP neighbors globally, include the keyword `import` *`policy-name`* configuration statement at the `[edit protocols rip]` level. Similarly, to have RIP advertise routes through policy, apply the keyword `export` *`policy-name`* statement at the global `[edit protocols rip]` level. If more than one policy name is included, they are evaluated first to last (left to right).

```
[edit protocols rip]
jncia@my.router# import policy-name

[edit protocols rip]
jncia@my.router# export policy-name
```

To selectively configure for a specific neighbor, add the keyword statement at the `[edit protocols rip group *group-name* neighbor *address*]` level.

For more information on JUNOS policy, see the *Policy* chapter.

Monitoring RIP in JUNOS

The basic commands to assist in troubleshooting and maintaining RIP are listed below.

```
jncia@my.router> show rip ?
Possible completions:
  general-statistics   Show RIP general statistics
  neighbor             Show RIP interfaces
  statistics           Show RIP statistics
```

```
jncia@my.router> show rip neighbor <interface>
```

This will display the status of current adjacencies. The interface and state are shown in the first columns. The configured modes for sending and receiving updates are noted in column five and six. Additionally, the metric applied to updates arriving on the interface will be shown in the final field.

```
jncia@my.router> show rip neighbor



                        Source          Destination     Send    Receive    In
Neighbor       State    Address         Address         Mode    Mode      Met
--------       -----    -------         -----------     ----    -------   ---
fe-0/0/0.2       Up    10.21.1.2        (null)          bcast   both        2
fe-0/0/0.1       Up    10.21.2.2        (null)          none    v2 only     1
```

```
jncia@my.router> show rip statistics <interface>
```

This command will display counters detailing update traffic for the specified interface. Statistics for all interfaces will be returned if no particular one is requested. Note the protocol timers listed on the line of output.

```
jncia@my.router> show rip statistics fe-0/0/0.2
```

```
RIP info:port 520; update interval 30s; holddown 180s; timeout
120s.
   rts learned  rts held down  rqsts dropped  resps dropped
            10              0              0              1

fe-0/0/0.2:  10 routes learned; 2 routes advertised
Counter                        Total   Last 5 min  Last minute
-------                   -----------  -----------   ----------
Updates Sent                      0            0           0
Triggered Updates Sent            1            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            0            0           0
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
```

Key Points

RIP is selected as it is relatively easy to configure and simple to operate. In light of its simplicity, this protocol takes significant coverage in the JNCIA exam. If for that reason alone, read this chapter carefully and be sure you can answer the test questions at the end.

- RIP is a distance vector protocol.
- Uses the Bellman-Ford algorithm to compute route preference.
- No authentication in Version 1
- RFC specifies plain-text passwords for Version 2
    - JUNOS supports plain-text and MD5 encrypted passwords in Version 2
- Metric is hop-count.
- Hop count of 16 is unreachable.
- Utilizes Poison Reverse and Split Horizon to prevent routing loops.
- Does not scale to large networks.
- Update timer defaults to 30 seconds.
- Dead timer defaults to 120 seconds.
- 25 destinations per update message (24 if authentication is configured).

Additional Information

For additional information, please consult the following at http://www.ietf.org:

- RFC 1058 Routing Information Protocol
- RFC 1721 RIP Version 2 Protocol Analysis
- RFC 2453 RIP Version 2

Chapter Five

OSPF

Chapter 5: OSPF

Overview

In this section you will learn about the dynamic routing protocol OSPF and its fundamental concepts. By the end of this chapter you should understand and be able to define:

- ✓ The purpose of Link-state routing protocols
- ✓ Route path selection in OSPF
- ✓ Operation of the Hello protocol
- ✓ Stages of neighbor adjacency
- ✓ Link-state Advertisements
- ✓ OSPF network types
- ✓ Area Types
- ✓ OSPF router types

Introduction

You have seen how the implementation of dynamic routing protocols assists network administrators and allows for increased complexity with the addition of RIP. However, rapid network expansion soon pushed RIP beyond practical limitations on large networks. *Open Shortest Path First* (OSPF) protocol was developed in response to a need in the networking community for a robust, non-proprietary *Interior Gateway Protocol* (IGP). As an IGP, OSPF routes packets within a single *Autonomous System* (AS) – the term used for a network under a common administration. It was designed for TCP/IP, and as such explicitly supports *Variable Length Subnet Masking* (VLSM) to make better use of address space. All versions of JUNOS software support OSPF version 2.

In addressing the most problematic issues with RIP, OSPF is a departure from it on several fundamental levels. Whereas RIP is a distance vector protocol relying on hop-count to determine best path selection, OSPF is a *link-state* protocol. Link-state protocols require that each participating node have full knowledge of the complete network topology. Each node (router in this case) must keep track of the link status, or state, of each of its connections and immediately notify the other nodes of any changes occurring.

Using link-state information to make routing decisions, OSPF utilizes the *Shortest Path First* (SPF) algorithm to determine the lowest cost link between two nodes in the same area, eliminating the limitation of hop-count. The SPF calculation is based upon the Djikstra Algorithm, named after Dutch scientist Edsgar Djikstra, which determines the distance between a matrix of nodes. Therefore, by applying a 'distance' or cost to each link on an internetwork the SPF algorithm can compute the shortest or least costly path between any two nodes.

Fundamentally, the SPF tree will calculate the cost by adding the link *metric* of all intervening connecting nodes. The metric is an administratively defined cost for the link leaving a router. By default, it is based on the interface bandwidth, but can be configured to reflect the desire of the administrator. It is important to remember that OSPF metrics will control which path packets take through the network, so it is best to rely on a consistent formula when deciding metrics.

In Figure 5.1 below, to get from node A to node E, OSPF will look at the cost to go from A to D and add that to the cost of going D to E. Likewise, it will also calculate the cost of going via B in a similar manner. When all possible paths have been accounted for, the lowest cost path is selected. It is also important to note that OSPF costs are assigned to the interface on the router rather than to physical link itself, so it is possible to have asymmetric costs, meaning the metrics could be different for each direction. This is illustrated as well in the below diagram (Figure 5.1) where the cost to go to router A **from** B is 50, while the cost from A **to** B is 20. It is possible for this to make instances of asymmetric routing, where traffic between two nodes takes different paths. If we were to assume that all of the unmarked paths have a cost of 10 and the links between A and B cost as marked, packets going from A to B will flow directly across their shared link while packets returning from B to A will travel B – E – D – A. This is because the interface connecting B directly to A costs 50 while each of the interfaces B to E, E to D and D to A cost 10, for a total cost of 30. Similarly, since the cost on A's interface to B is 20 and the route through D-E costs 30, it is the lowest cost path.

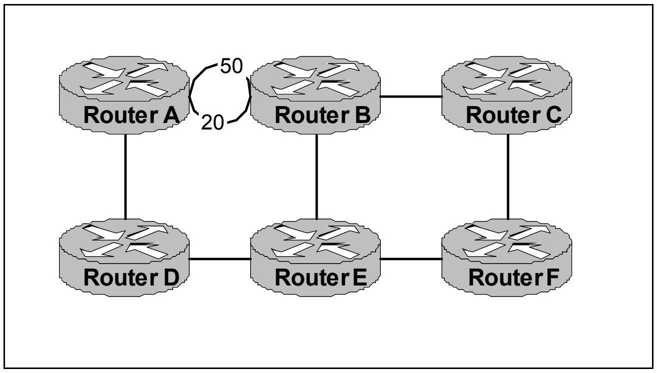


Figure 5.1 OSPF Matrix and Metrics

If there are multiple links with the same overall cost available to get to a specific destination the Juniper router will randomly select a next-hop. If the next hop changes at any time, the route selection will be redone in a random fashion. In the above diagram, for example, packets leaving Router C destined for Router E have two equal cost paths: one next hop on Router B and another through Router F. Because of this, JUNOS will randomly select which next-hop to install as the active route.

Adjacencies

In order for routers running OSPF to exchange data, there must be a mechanism for them to determine other nearby, or adjacent, nodes with which they can communicate. To enable OSPF on a router, you must explicitly declare which interface(s) will be taking part in the OSPF process. Each router must also supply a *Router ID* (RID) to uniquely identify it. The RID is typically a loopback address, but can theoretically be any unique IP address allocated to the router. Upon declaring active interfaces and the RID, the router is ready to begin communicating with the other routers on its directly connected networks. OSPF implements a *hello* protocol to establish and maintain adjacencies with neighboring routers.

Hello Protocol

Through the exchange of *hello* packets OSPF routers establish and maintain adjacencies. However, simply sending and receiving *hello* packets out active interfaces is not enough to bring up a stable adjacency. There are a number of configurable parameters that must match between connected nodes before an adjacency will form.

Upon starting up, OSPF will send *hello* packets out every active OSPF interface to the special multicast address of 224.0.0.5. All OSPF routers listen for packets with this address. The *hello* packets include the source and destination IP addresses, the RID, and the configured values for the *Area ID*, authentication, network mask, *Hello Interval* and *Router Dead Interval*. If any of the configured values do not correspond between routers, an adjacency will not form. The Hello Interval is the number of seconds between *hello* packet advertisements. The default value is 10 seconds. The Router Dead Interval is the amount of time that can pass without receiving a *hello* packet before a router will consider a neighbor unreachable. The default router dead interval (also known as the dead or hold-down timer) is 40 seconds. *Hello* packets reside on top of IP, utilizing port 89.

As an OSPF router sends out and receives hello packets, it goes through the below stages in forming an adjacency:

- **Down** – The first state of adjacency. This means that no information has been received from the neighbor. If an OSPF router fails to receive a Hello packet from a working neighbor within the Dead Interval (normally four times the Hello interval), its state changes to DOWN.
- **Init** (Initialize) – This state reflects that the router has received a Hello packet from a neighbor, but it does not see its own RID in the *hello*. When a router receives a *hello* packet from a neighboring router, it includes the sender's router ID in its own hello advertisements.
- **2-Way –** This state indicates that the routers have established communication in both directions. Since an OSPF router includes the RID of a received h*ello* in its own h*ello* packet, a router receiving a h*ello* packet with its own RID included within it knows the neighbor is receiving its advertisements. Routers that are not DR/BDR on a BMA network will remain in '2-way' (see *Network Types*).
- **Exstart** (exchange start) **–** This is the first step in forming a legitimate adjacency. A *master/slave* relationship is determined at this point for the actual database exchange.
- **Exchange –** Routers exchange *link-state advertisements* (LSAs) in this state with *database description* (DBD) packets. Each LSA

update contains a sequence number assigned by the master and is explicitly acknowledged by the slave router.

- **Loading –** During an adjacency, if a router receives a missing or out of date LSA it requests an update by sending a link-state request. This state indicates the router is in the process of updating its LSA database.
- **Full –** Signifies the OSPF databases are synchronized between neighbors. Routers in *full* are fully adjacent. When all neighbors in an area are fully adjacent to their neighbors the area is considered *converged*.

The behavior of the *hello* protocol varies slightly depending upon the type of network over which it is working. There are four different types of OSPF networks, listed below.

Network Types

There are 4 different network types on which the JNCIA tests knowledge:

- Point-to-Point (P2P)
- Broadcast Multi-access (BMA)
- Non-Broadcast Multi-access (NBMA)
- Point-to-Multipoint (P2MP)

Point-to-Point

This is the simplest of the above network types. Two routers are directly connected to each other, which means that each router only has a single neighbor over that interface. These links are normally configured over serial or SONET connections. Neighbors on a point-to-point link should be in *full* state.

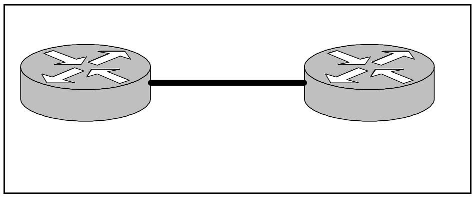


Figure 5.2 Point to Point Network

Broadcast Multi-Access Networks (BMA)

Broadcast networks are *local area network* (LAN) connections between routers that are capable of broadcasting. Ethernet is by far the most common type of this connection. For a full mesh, each router on a multi-access network would need one less than N neighbors, where N is the total number of nodes on the broadcast network running OSPF. If there are five nodes running OSPF on an Ethernet segment, *each node* has 4 neighbors. To reduce the amount of routing overhead between neighbors on the segment, OSPF elects a node to be the primary communicator and another to be the secondary. These are called the *designated router* (DR) and *backup designated router* (BDR), respectively.

All nodes on the BMA form a Full adjacency with the DR and BDR only. The remaining neighbors stay in 2way state, meaning they do not send or receive updates from those routers. It is the DR's primary responsibility to ensure that all routers connected to the broadcast network receive updates about all changes to the BMA topology. It is the job of the BDR to make certain the DR does not fail in delivering updates to the network. If the BDR senses that the DR has failed, it assumes the role of DR and distributes the routing updates itself. For this reason, the BDR and DR have synchronized link-state databases. For more on the rules to determine DR/BDR election, see the following section *Multi Access Networks and Designated Router Elections.*

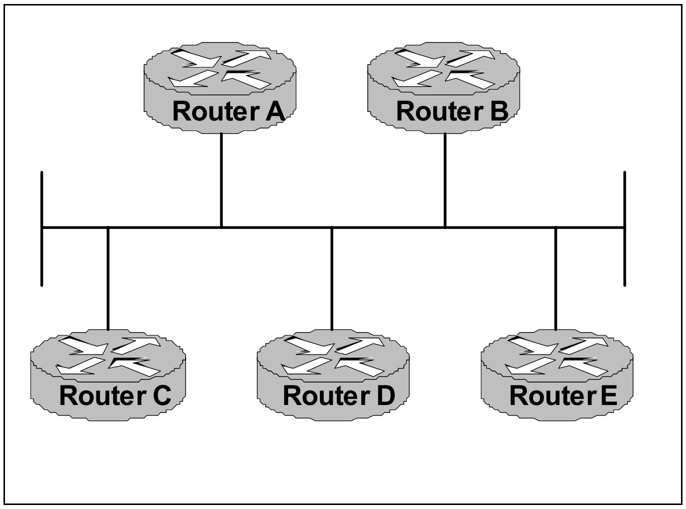


Figure 5.3 Broadcast Multi-Access Network

Non-Broadcast Multi-Access (NBMA)

This type of network connects multiple routers across a non-broadcast medium, such as X.25, ATM, or frame relay. These types of connections have intermediate switches for virtual circuit termination placed between routers. Default behavior for an NBMA network requires a designated router be established. The routers connected to a NBMA network can be partially or fully meshed.

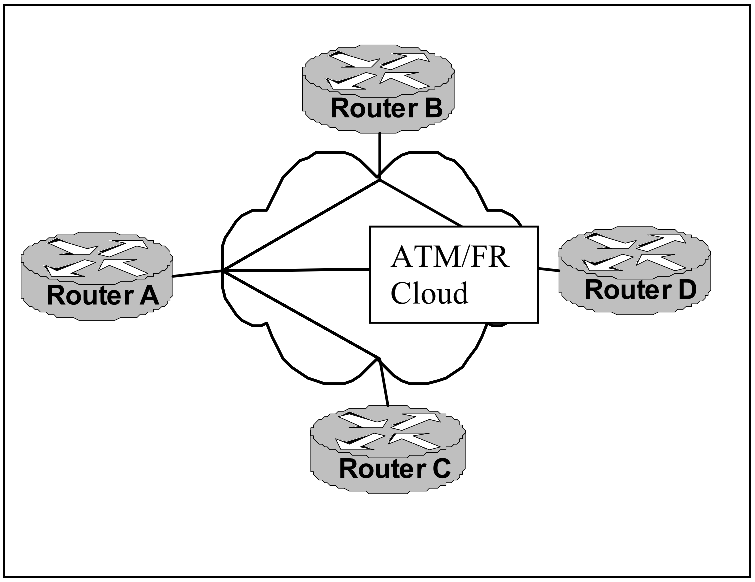


Figure 5.4 Partial Mesh NBMA or P2MP Network

NBMA networks have one additional state when forming adjacencies called *attempt*. While in this state, the router is attempting to send *hello* packets to the neighbor but has not yet received any information. The *attempt* state is only valid for NBMA participating routers and will not be seen on other network types.

Point to Multipoint (P2MP)

From the perspective of the JNCIA exam, this network type is identical to the NMBA with the exception of a full network mesh. It can be treated as a collection of P2P networks. There are no DR or BDR routers required.

Multi-Access Networks and Designated Router Elections

As discussed above under BMA networks, a full mesh between routers on a multi-access network would require each router to have one less than N neighbors where N is the number routers on the segment. It is the job of the DR/BDR system to ensure all neighbors receive network topology updates while reducing the amount of traffic on the attached network. To do this, during the initial phase of adjacency forming OSPF routers will elect a Designated Router and a Backup Designated Router. Neighbors will only form a full adjacency with the routers that are selected to be DR/BDR and will remain in 2way with the other routers on the segment.

If only one router is up on a multi-access network, it is the DR. When a second router is added, it becomes the BDR. Subsequent routers will form adjacencies with the DR/BDR but remain in 2-way with one another. No elections take place until either the DR or BDR fail. An election takes place if the DR fails. The BDR becomes the new DR and a new BDR is elected.

OSPF uses router priority to determine which routers are the most preferred for DR/BDR election. Router priority is a configurable number from 0 to 128, where a higher value is more preferred to become DR. A priority of 0 has special significance. A router with priority 0 **cannot** become DR. If multiple routers have the same priority, as they would by default, then the lower Router ID is used. Remember, if a router with a higher priority comes up on a LAN segment that already has a DR it **will not** pre-empt or force an election, but will be elected if the acting DR fails in the future.

Area and Router Types

OSPF is a hierarchical protocol. To reduce the size of the link-state database and the frequency of LSA flooding to all nodes, an OSPF domain may be broken into different logical areas which function independently of one another. Each node running OSPF in the area must have an identical link-state database. Nodes in different areas may have differing databases.

Area 0

Area 0 is the backbone area and is the fundamental area. If there is only one area in an OSPF domain, it can be any number. Most often, for reasons of simplicity and scalability, it is configured as area 0. A simple illustration of this is in Figure 5.5.

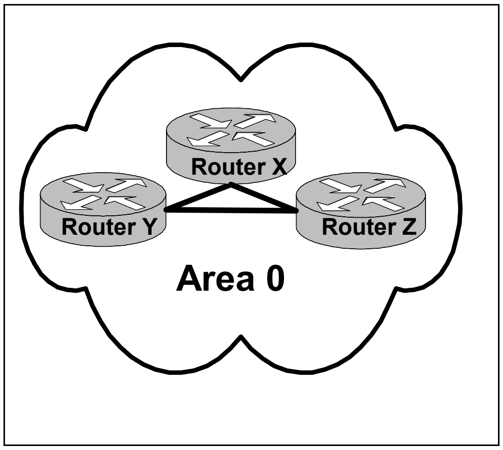


Figure 5.5 Single Area OSPF Map

We could technically call Routers X, Y, and Z backbone routers, but there is only one area and therefore no "true" backbone. All of these routers have the same database of information regarding the area.

Site Areas

In multi-area configurations, the backbone is used to transport data between site areas. Traffic sourced from one area destined for another must transit the backbone area and all site areas must connect to area 0 (there is one exception to this rule and it will be discussed under *Virtual Links*). Sites are identified by an *Area ID.* Obviously, for OSPF to work correctly, all routers in a specific area must share the same Area ID. The Area ID usually follow two formats; a positive integer (such as 123) or a dotted decimal format (0.0.0.0 is a legitimate backbone ID). JUNOS will automatically convert regular integers to the dotted decimal format in the router configuration. If we grow our network to attach four site areas to the backbone of figure 5.5, we end up with the following:

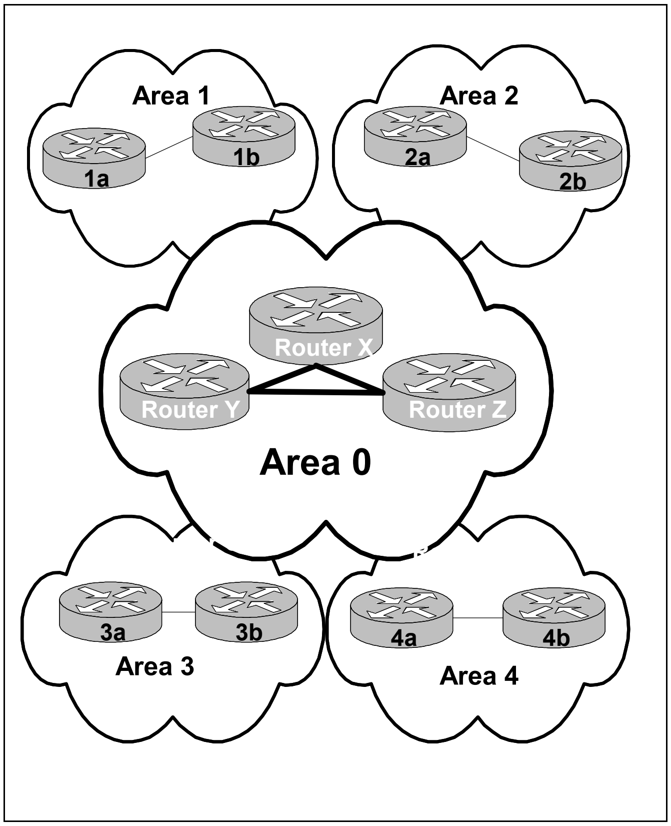


Figure 5.6 Multiple Area OSPF Map

We now have five areas in our OSPF domain. Each of the routers has a complete view of how to get to everything **in its own area**. Routers in area 1 (those named 1a and 1b) have no idea how to get to any destination in area 2, or any other area for that matter. Indeed, without adding some additional elements, none of these routers are aware other areas even exist!

Communicating between Areas

To communicate between areas, OSPF relies on routers that have interfaces in more than one area. These routers are called *area border routers* (ABRs). For instance, if a router were to have an interface connected to area 0 and an interface into a site area, it would be an *area border router* and would allow traffic to flow between the site and backbone. ABRs maintain a separate database for each area to which they are connected.

By growing our network once more, adding ABRs to allow inter-area traffic flow, our network diagram becomes rather more complex. However, remember that this is nothing more than the culmination of adding simple areas together.

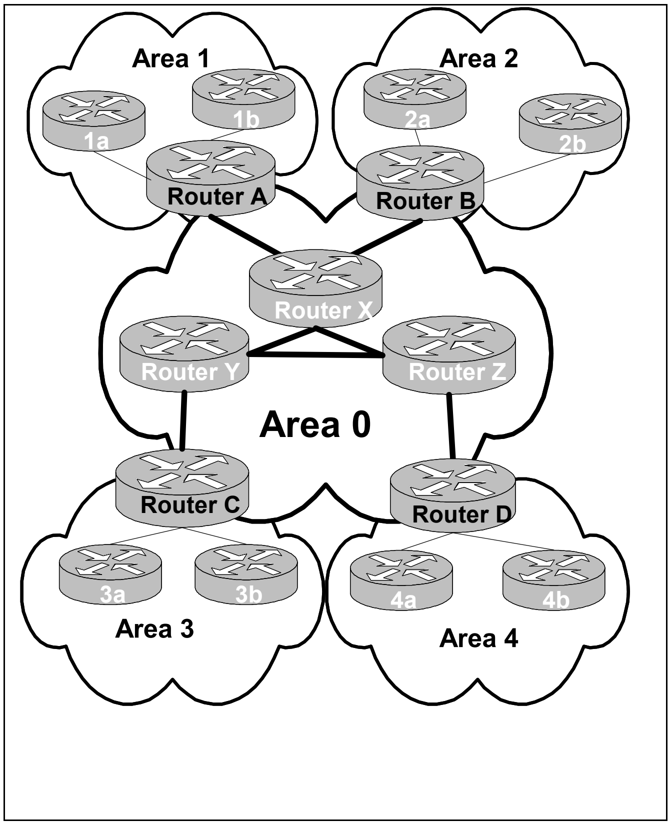


Figure 5.7 Multiple ABR OSPF Map

Now we can see that Routers A, B, C, and D are all ABRs with one interface in the backbone area and another in a site area. Site internal routers will learn about other areas and know to pass their traffic to their ABR to get there. Traffic flowing from Area 4 to Area 1 will pass through ABR D onto Backbone Router Z, to Backbone X, and to ABR A. In this way as well, each of the areas is insulated from route churn and instability within the other areas. Destination routes flapping in Area 1 will not cause the routers in Area 4 to re-run SPF calculations. This is another reason why large networks are broken up into smaller, more manageable site areas.

Routers that exchange routes with other ASs are called *AS boundary routers* (ASBRs). They advertise AS external routes to the entire OSPF domain. Note that the definition of AS within OSPF differs from that of BGP. Any router, regardless of whether it is a backbone router, area internal router, or ABR may be an ASBR. An example of an ASBR would be a router redistributing RIP routes into OSPF. The RIP routes are considered external to the OSPF process and when redistributed will originate from the ASBR.

Types of Areas

In addition to the different types of routers and networks, there are multiple different types of site areas in OSPF. A simple site area has no additional stipulations on its behavior. A *stub area* is one in which AS external announcements are not flooded. This would be beneficial if the number of external routes coming in is large and updates could be made much more compact by their exclusion. An ABR facing a stub area will automatically announce a default route in place of the specific external routes so the site routers can still reach those prefixes. A stub area cannot contain an ASBR, so you cannot redistribute from another protocol into the stub area. Additionally, a stub area may not contain a virtual link.

A *not-so-stubby area* (NSSA) allows an ASBR to be a member of the site area. Both NSSA and stub areas do not allow external routes to be flooded into the area. However, an NSSA allows for the injection of external routes which can then be flooded within that area and outward to the backbone. In effect, the NSSA can redistribute routes from another protocol but receives a default route for external advertisements outside of its native area.

Virtual Links

Ideally, all site areas should have a direct connection to Area 0. There is one exception to this rule: *Virtual links*. It may be necessary at some point that a site area is not directly connected to area 0. By configuring a virtual link, it is possible to create a logical tunnel between area 0 and the removed site area by transiting an intermediate site area. Adding virtual links generally adds unnecessary complexity and increases the possibility of area failure and should be considered a temporary fix rather than a permanent solution. Virtual links cannot be configured stub areas.

In figure 5.8 Area 7 is connected to Area 0 via Area 6. Should the transit link or Area 6 go down, Area 7 will be disconnected from the backbone as well.

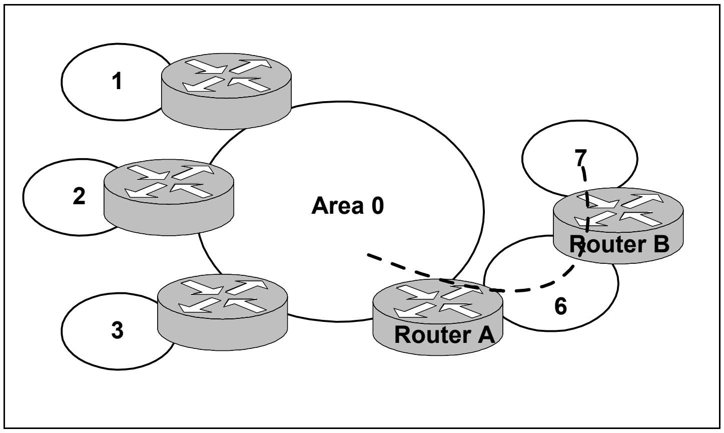


Figure 5.8: Virtual Link to Area 0

Link-state Advertisements (LSAs)

Routers participating in OSPF communicate to one another through the use of Link-state Advertisements (LSAs). LSAs contain the information OSPF routers require to build and maintain a view of the area they are in. The *hello* protocol determines how OSPF itself talks to neighbors and maintains adjacencies. LSAs are behind the workings of how OSPF determines where to route traffic.

Each OSPF router maintains a topology database for the area. It is absolutely necessary that these tables be in agreement, or synchronized, to prevent routing loops. OSPF routers flood LSAs throughout the network to maintain consistent route topologies between nodes. For this reason, adding import policy to OSPF is frowned upon as it has the potential to produce inconsistent topologies within areas. When there is a topology change, LSA flooding ensures that all OSPF databases converge quickly and accurately. There are five commonly used LSA types that will be covered on the JNCIA examination. Other types of LSAs exist for different services, however the JNCIA will not require knowledge of these.

- ❖ Type I Router LSA – Information about the router and its directly connected links. Type I LSAs are flooded only within the area.

- ❖ Type II Network LSA – Information about a LAN and the routers connected to it. These LSAs are advertised by the DR and are only flooded into the site area to which it is a member.

- ❖ Type III Summary LSA – Originated from the ABR, these describe networks that are reachable outside each of the ABR's areas.

- ❖ Type IV ASBR Summary LSA – Define routes to the ASBR. Type IV originate on the ABR.

- ❖ Type V External LSA – Include information about destinations outside the OSPF domain (or AS). They originate from an ASBR and are flooded throughout the entire OSPF network.

The types of LSAs that a router will announce depend upon the type of role it plays. In other words, whether or not the router is a DR, BDR, ABR or ASBR will determine the mix of LSAs that it will choose to flood.

OSPF will exchange its entire database when two neighbors initialize an adjacency. All triggered updates contain only information regarding which routes have changed. Updates, therefore, are much smaller and consume less overhead, even in large networks.

<u>OSPF Configuration within JUNOS</u>

Now that we have covered the theory behind OSPF, we can examine which steps are necessary to configure OSPF on a Juniper router. The fundamentals that should be remembered for the JNCIA exam include:

- OSPF configuration takes place under *protocols ospf* in edit mode
- The minimum OSPF configuration consists of enabling the protocol on a global level, and defining the interfaces that will participate and the area which they will be in
- The default timers for OSPF are 10 seconds for the *hello interval* and 40 seconds for the *dead timer*

The remaining options that can be configured depend heavily upon the type of network the interface will communicate on. We covered the fundamentals of network type previously. The JNCIA examination does not focus on the configuration of OSPF, rather the real emphasis is on understanding the behavior of the protocol and what problem indicators to look for. Nevertheless, there are some potential exam areas within basic OSPF configuration.

The top level of configuration for OSPF is under 'edit protocols ospf'. All global and area specific commands are set at this tier. Configuring OSPF on a router requires defining a minimum of two things:

1) Which interfaces will participate in OSPF.
2) Which areas those interfaces will be assigned to.

The minimum required configuration follows the form of:

```
[edit protocols]
jncia@my.router# show
        ospf {
                area 0.0.0.0 {
                        interface interface-name;
                }
        }
}
```

When multiple areas are defined, area 0 is called the backbone area. To add additional interfaces into the backbone area is a simple addition to the minimum configuration:

```
[edit]
jncia@my.router# set protocols ospf area 0.0.0.0
interface interface-name
```

The configuration for a site area is similar to the minimum, or backbone, configuration, the only exception being that the area ID is some number other than 0.

```
[edit protocols ospf]
jncia@my.router# set area area-id interface
interface-name
```

(Note that in the above we are already at the [edit protocols ospf] level of configuration)

Both Stub and NSSA are configurable under the `area` level of the edit hierarchy. An area cannot be both a Stub and an NSSA.

```
[edit protocols ospf area area-id]
jncia@my.router# set [stub|nssa]
```

Metric

All OSPF active interfaces have a cost associated with them that is used as the routing metric when calculating the shortest path. Unless changed, the metric is calculated by a ratio of the reference bandwidth to the interface bandwidth. By default, the reference bandwidth is 100Mbps (100,000,000bps). There are then two ways to modify the OSPF metrics. To change the metric for routes advertised from a specific interface include the `metric <cost>` statement at the `[edit protocols ospf area` *`area-id`* `interface` *`interface-name`*`]` level. The metric can be a positive integer from 1 to 65535.

```
[edit protocols ospf area 0.0.0.0 interface so-4/0/0.0]
jncia@my.router# set metric cost cost
```

Authentication

Exchange of OSPF packets between devices can be configured to use authentication, ensuring that only trusted routers participate. By default, authentication is disabled; however, one of the following types can be enabled.

- Simple authentication - uses a plain text password.
- MD5 algorithm - utilizes MD5 encryption to provide a stronger level of security.

Each router in the area must use the same type of authentication. Both types of authentication are passed in the transmitted h*ello* packets and are verified with a key by the receiving router. Configuring authentication is done under the level of edit protocols ospf area:

```
[edit protocols ospf area area-id]
jncia@my.router# set authentication-type
[none|simple|md5]
```

If needed, the key is then configured beneath the interface hierarchy level:

```
[edit protocols ospf area area-id interface
interface-name]
jncia@my.router# set authentication-key <key> key-id <id>
```

The key, or password, may be from one to eight ASCII characters in length. As throughout JUNOS, it is required to enclose the entire string in quotes if it includes spaces.

Configuring OSPF Timers

In order to maintain an accurate link-state database, OSPF routers send and expect to receive *hellos* and LSAs at specific intervals. In certain situations, the network administrator may wish to change these intervals from the default value. It is important to remember however, that the timer intervals need to be consistent across the area to establish working adjacencies.

Hello packets are sent to establish and maintain adjacencies with neighboring routers on every active OSPF interface. The default *hello timer* is 10 seconds.

The time that a router will wait before declaring a neighbor unreachable is called the *router dead interval*. It is four times the hello interval, 40 seconds by default. It can be modified similar to the *hello interval* under the same edit level:

```
[edit protocols ospf area area-id interface
interface-name]
jncia@my.router# set hello-interval seconds
```

```
[edit protocols ospf area area-id interface
interface-name]
jncia@my.router# set dead-interval seconds
```

Policy in OSPF

Similar to the way in which policy is applied in RIP to filter imported and exported routes, we have the option of configuring policy for OSPF. However, it should be noted that import policy applied to a link-state protocol like OSPF or IS-IS can lead to inconsistent databases and topologies. As a rule, only consider export policy as an acceptable option for OSPF.

```
[edit protocols ospf]
jncia@my.router# export policy-names
```

For additional information on syntax and policy flow, see the *Policy* section.

Monitoring OSPF in JUNOS

The basic commands to ease troubleshooting problems with OSPF are listed below. Problems with OSPF running correctly can be related either to improper configuration or with network occurrences, such as failed circuit connections or router hardware. Optional parameters are enclosed in [brackets]. Tokens that should be replaced with actual interface or address information are enclosed in <angle-brackets>.

```
jncia@my.router>  show ospf ?
Possible completions:
  database        Show the OSPF link-state
  database
  interface       Show OSPF interface status
  io-statistics   Show OSPF I/O statistics
  log             Show OSPF SPF log
  neighbor        Show OSPF neighbor status
  route           Show the OSPF routing table
  statistics      Show OSPF statistics
```

```
jncia@my.router> Show ospf neighbor <neighbor
address> <brief|detail>
```

If the neighbor address is left out, an abbreviated list of neighbor information will be shown:

```
jncia@my.router> show ospf neighbor
  Address      Interface     State      ID            Pri  Dead
10.1.11.53     so-0/0/0.0    Full       10.2.8.241    128    39
10.1.11.58     so-0/1/0.0    Full       10.2.44.98    128    36
10.1.27.66     ge-7/0/0.0    Full       10.2.44.99      1    36
10.1.27.130    ge-7/0/1.0    2way       10.2.44.96      1    31
```

The 'show ospf neighbor' output is very useful for determining the state that neighboring routers are in. From here we can see the address of the neighboring interface, the local interface name, the State the adjacency is currently in, the RID of the neighbor, the Priority, and the Dead-Timer. Ideally, we expect neighbors to be in *Full* if they are P2P, or *2way* if they are on a BMA network. States other than these usually indicate a problem. Including the neighbor address will allow inspection of that adjacency alone. Adding 'detail' will give additional information.

```
jncia@my.router> show ospf neighbor 10.1.27.66 detail
Address        Interface     State     ID             Pri  Dead

10.1.27.66     ge-7/0/0.0    Full      10.1.44.99        1   36
  area 0.0.0.33, opt 0x42, DR 10.1.27.65, BDR 10.1.27.66
  Up 14w1d 21:36:45, adjacent 14w1d 21:36:45
    Link-state retransmission list:  4 entries
```

Note that the 'detailed' output also now displays the DR/BDR *interface* address and area information. This router is in area 33. The DR for this segment is 10.1.27.65, which just happens to be this router's interface. The BDR for the segment is this neighbor, 10.1.27.66. This neighbor has been up for over 14 weeks.

```
jncia@my.router> Show ospf interface <interface-name> <brief|detail>
```

If the interface name is left out, an abbreviated list of neighbor information will be shown:

```
jncia@my.router> show ospf interface
Interface   State    Area      DR           IDBDR ID   Nbrs
lo0.0        DR      0.0.0.0   10.2.44.97   0.0.0.0       0
so-0/0/0.0 PtToPt    0.0.0.0   0.0.0.0      0.0.0.0       1
so-0/1/0.0 PtToPt    0.0.0.0   0.0.0.0      0.0.0.0       1
ge-7/0/0.0   DR      0.0.0.33  10.2.44.97   10.2.44.99    1
ge-7/0/1.0   DR      0.0.0.33  10.2.44.96   10.2.44.97    3
```

From the output of 'show ospf interface' we can see which local interfaces are OSPF active, what State they are in, which Area they belong to, what the DR/BDR RIDs are and how many neighbors are adjacent. It is important to note that the DR/BDR information listed here is the *Router ID* whereas that listed under ‘show ospf neighbor x.x.x.x detail’ is the DR/BDR *interface* address. Note that an interface will still show up with this command even if it has no neighbors because it is still an OSPF active interface. Two reasons for this lack of neighbors could be:

1) The interface in question is configured as passive.
2) The neighbors on that interface are down.

```
Show ospf database area <area-id>
```

```
jncia@my.router> show ospf database area 0.0.0.33

    OSPF link-state database, area 0.0.0.33
Type       ID              Adv Rtr          Seq         Age  Opt  Cksum   Len
Router    *10.2.44.97      10.2.44.97       0x8000798b   952  0x2  0xa0ab   36
Router     10.2.44.98      10.2.44.98       0x80007a2b   317  0x2  0xaef2   36
Router     10.2.44.99      10.2.44.99       0x80003c3b  1142  0x2  0xf6dc   60
Router     10.2.44.100     10.2.44.100      0x800038d2   629  0x2  0xf91c   60
Network   *10.2.27.65      10.2.44.97       0x800037fb   956  0x2  0x61a7   32
Network    10.2.27.69      10.2.44.98       0x80002f78   321  0x2  0x6a23   32
Network    10.2.27.74      10.2.44.100      0x800038c3   629  0x2  0x9e72   32
Summary   *10.6.1.1        10.2.44.97       0x800001bd  1376  0x2  0x3ece   28
Summary    10.6.1.1        10.2.44.98       0x800001be  1379  0x2  0x920    28
Summary   *10.6.1.2        10.2.44.97       0x800001b3   838  0x2  0x3747   28
Summary    10.6.1.2        10.2.44.98       0x800001b3   837  0x2  0x497    28
```

This command shows the actual link-state database that OSPF assembles from received LSAs. Using the command without an <area> identifier will output the entire database, while including the ID will list just the area in question. The first column indicates the type of LSA. The database is normally arranged beginning with Type I LSAs first. Entries marked with an asterisk represent those originating from the local router. As can be seen in the above, 10.2.44.97 is the local router, and the LSAs marked with * are local to this router. Note as well that more than one device can report the same route in an LSA, as with 10.6.1.2 and 10.6.1.1 above.

Below is the output taken from the OSPF log file illustrating the steps undertaken while an adjacency forms.

```
1) jncia@my.router>show log ospf-log | grep 172.16.1.1
2) OSPF neighbor 172.16.1.1 (fe-0/0/0.2) state changed
from Down to Init
3) OSPF neighbor 172.16.1.1 (fe-0/0/0.2) state changed
from Init to 2Way
4) RPD_OSPF_NBRUP: OSPF neighbor 172.16.1.1 (fe-0/0/0.2)
state changed from Init to 2Way
5) OSPF neighbor 172.16.1.1 (fe-0/0/0.2) state changed
from 2Way to ExStart
6) OSPF neighbor 172.16.1.1 (fe-0/0/0.2) state changed
from ExStart to Exchange
7) OSPF neighbor 172.16.1.1 (fe-0/0/0.2) state changed
from Exchange to Full
8) RPD_OSPF_NBRUP: OSPF neighbor 172.16.1.1 (fe-0/0/0.2)
state changed from Exchange to Full
```

Line 1 indicates we are looking into the logfile “ospf” and using the | to include a UNIX style grep. This limits output to match only lines including “172.16.1.1”. Neighbor 172.16.1.1 begins in the “Down” state. From line 2 we can see the state change to “Init”. The rest of the output illustrates the process of building a ‘Full’ adjacency as it moves through the remaining stages.

Key Points

OSPF is a robust and scalable link-state protocol that rapidly converges. With the information supplied in this chapter you should be able to understand the workings of the protocol, configure simple OSPF neighbors and areas, and troubleshoot everyday problems that may arise.

- OSPF is a link-state protocol. Link-state protocols require that each participating node have full knowledge of the complete network topology. Each router must keep track of the link-state of each of its connections and immediately notify the other nodes of any changes occurring.
- Utilizes the Shortest Path First (SPF) algorithm to determine the lowest cost link between two nodes in the same area. The SPF calculation is based upon the Dijkstra Algorithm. There is no hop-count.
- Each OSPF router maintains a topology database for the network. It is absolutely necessary that these tables be in agreement (are 'synchronized'), to prevent routing loops.
- OSPF is a hierarchical protocol. An OSPF domain may be broken into different independent logical areas. Area 0 is the backbone area.
- Interfaces running OSPF form adjacencies with neighboring interfaces through the use of Hello packets.
- OSPF routers flood Link-state Advertisements (LSAs) throughout the network to maintain consistent route topologies between nodes. When there is a topology change, LSA flooding ensures that all OSPF databases converge quickly and accurately.
- Types of Link-state Advertisements:

    - Type I Router LSA – Information about the router and its directly connected links. Type I LSAs are flooded only within the area.
    - Type II Network LSA – Information about a LAN and the routers connected to it. These LSAs are advertised by the DR and are only flooded into the site area to which it is a member.
    - Type III Summary LSA – Originated from the ABR, these describe networks that are reachable outside each of the ABR's areas.
    - Type IV ASBR Summary LSA – Define routes to the ASBR. Type IV originate on the ABR.

- Type V External LSA – Include information about destinations outside the OSPF domain (or AS). They originate from an ASBR and are flooded throughout the entire OSPF network.

Other types of LSAs exist for different services; however, the JNCIA will not require knowledge of these.

Additional Information (RFCs)

For additional information, please consult the following at http://www.ietf.org:

- RFC 1245: OSPF Protocol Analysis
- RFC 1246: Experience with the OSPF Protocol
- RFC 2328: OSPF Version 2
- RFC 1557 NSSA
- RFC 1584 MOSPF
- RFC 2740 IPv6

Chapter Six

IS-IS

Chapter 6: IS-IS

Overview

In this section you will learn about the dynamic routing protocol IS-IS and its fundamental concepts. By the end of this chapter you should understand and be able to define:

- ✓ The origin and general purpose of IS-IS.
- ✓ NSAP addressing.
- ✓ Purpose and configuration of domains, areas and levels.
- ✓ How IS-IS devices communicate.
- ✓ Some key differences between IS-IS and OSPF

Introduction

Intermediate System to Intermediate System (IS-IS) is an Open System Interconnection (OSI) link state Interior Gateway Protocol (IGP) designed around the International Organization for Standardization (ISO) concepts for networking. In the OSI model, networks are built around 'systems'; a router is an Intermediate System (IS) and an end host or user is an End System (ES). From this, we can tell that IS-IS deals with traffic flow between routers.

Similar to OSPF, IS-IS utilizes the Dijkstra algorithm within a Shortest Path First (SPF) calculation to compute the lowest cost path to a particular node. An IS-IS active router compiles a database filled with the link state information of all the other routers in its area. Part of this information is the cost of every particular link. SPF can then construct a logical tree with itself as the root that branches to every other node.

IS-IS was originally intended to route *Connectionless Network Protocol* (CLNP). It was extended to support IP, but its roots in OSI often make it appear more complex and difficult than it actually is.

IS-IS Logical Configuration

IS-IS domains are divided into areas to help control the amount of route information that needs to be distributed between nodes. Levels are used to designate a router's function within the area(s) to which it belongs. Interfaces on each router in the area are assigned to a level. Within JUNOS, each individual interface, once enabled to run IS-IS, can be placed into one, or both, of these levels.

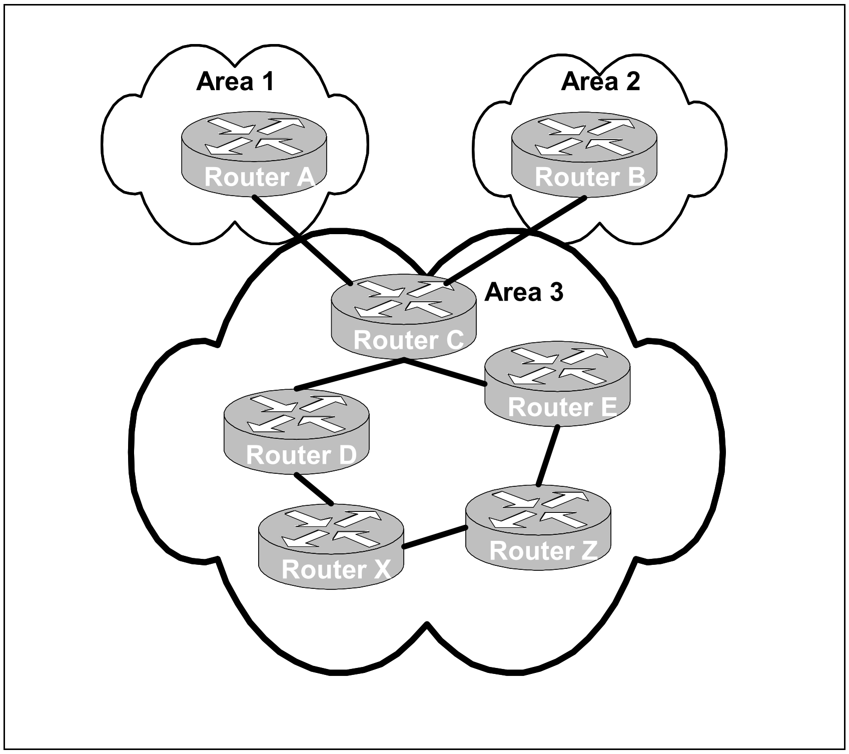


Figure 6.1: IS-IS Area Map

As can be seen in the figure 6.1, the area breakdown resembles OSPF logical areas except for one critical point, the lack of an Area 0. Of course, it would be possible to number an Area as 0 in IS-IS, but the functionality of a dedicated backbone area is not the same. In this sense, IS-IS differs from OSPF in that it does not treat inter-area routing in the same manner.

In addition to breaking the domain into areas, IS-IS functions are divided between two levels of responsibility. Level 1 systems route only within their respective area. When a Level 1 node encounters a destination that is outside of the Level 1 area, it forwards the route towards the nearest Level 2 system.

Level 2 systems route between areas and toward other autonomous systems. In a simplified manner, think of Level 2 routers as the backbone routers and Level 1 as the site routers. Be aware that the line is not always so clear and many factors play a role.

Using the same simple topology as we did previously and now appointing the various routers Levels according to their responsibilities, we get the somewhat more complex diagram:

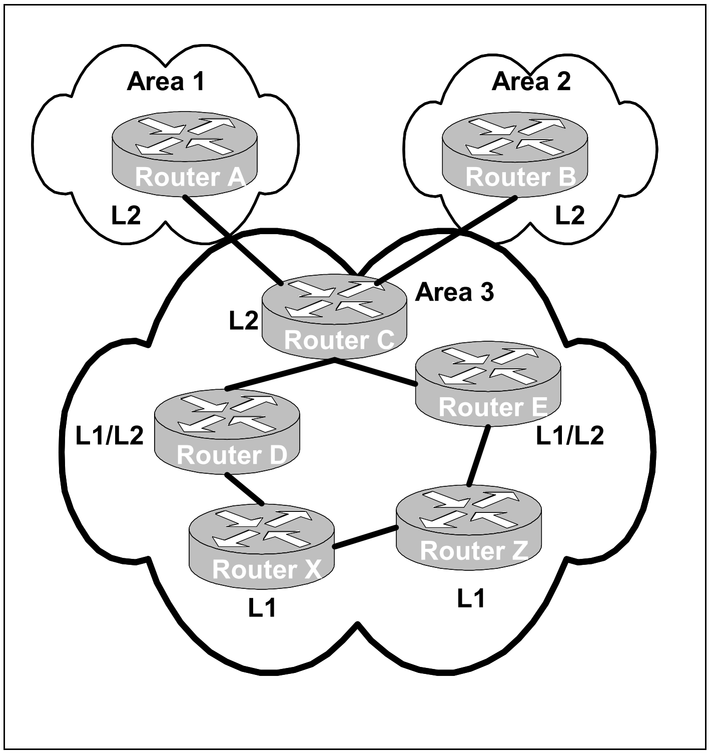


Figure 6.2: IS-IS Layer and Area Map

From this, we can see how the functions of the routers in the areas are broken down. Router X and Router Z are pure Level 1 devices. They have complete knowledge of how to get to all nodes in Area 3. However, if either needs to send traffic to Area 1 or 2, it needs to be sent outside their own local area. Typically, Level 1 routers will send traffic destined for another area to the closest L1/L2 router.

Routers D and E are Level 1 and 2 devices. They have interfaces in both levels and have knowledge of how to get to other areas. In this example, the path on to Area 1 and 2 is through Router C.

Router C is a pure Level 2 device. It is connected to other Level 2 devices, and it knows how to get to Area 1 and 2. Note that Router C is still in Area 3, but its purpose is to route packets on to other L2 or L1/L2 nodes. Similarly, Router A represents the Level 2 router within Area 1 and Router B performs this duty for Area 2. We can assume that Area 1 and 2 contain their own topologies made up of more L1/L2 and L1 devices. Otherwise, there would be no need to send traffic there.

All IS-IS enabled interfaces (with the ISO protocol family enabled) are by default placed into both Level 1 and Level 2 areas. Interfaces can have one or both levels removed via the disable command.

Unlike OSPF, which has a default metric based upon link bandwidth, IS-IS has a flat default metric of 10. This is important to keep in mind, as when first enabling IS-IS all links out of a router will have a metric of 10 regardless of the actual interface bandwidth and can result in sub-optimal routing. The metric may be configured to custom values by the administrator.

To provide for scalability in a large broadcast environment, IS-IS requires the appointment of a *Designated Intermediate System* (DIS) per broadcast segment. This allows updates to be sent to one router for re-propagation instead of a flooding by every node on the broadcast medium. This function is similar to a Designated Router in OSPF, but varies in that there is no backup assigned.

The IS-IS speaker with the highest priority is declared the DIS. JUNOS sets priority to a default of 64. In the event of a tie in priority during election, the highest MAC address wins. As noted, unlike OSPF, there is no backup DIS. The timers, as discussed shortly, vary when communicating with the DIS and allow any router to take this role should the DIS become unavailable.

ISO Network Addressing

Similar to the manner in which OSPF uses IP for network addressing, IS-IS uses OSI addressing. Each unique OSI address represents a network connection, such as a host interface or a router. A specific point of connection like this is identified as a *network service access point* (NSAP). A node may have multiple NSAP addresses. If this is the case, each of these addresses will differ only in the last byte (or the *n-selector*). Each network element is configured with a special NSAP address called the *Network Entity Title* (NET). Most systems have one NET. However, if they take part in multiple areas, they will have multiple NETs. The identification of the NET is made easier in that its last byte, or *n-selector* (NSEL), is 00.

One very important difference between IP and NSAP addressing to keep in mind is that while an IP address is likely to be configured on each interface, a single NSAP is used to address the entire router. Because the only address used is for the router itself, that address is the NET and is normally assigned to the router loopback. As the JNCIA deals only with routers, valid NET/NSAP will end with NSEL 00.

OSI addressing is a departure from the IP addressing scheme with which most people are familiar. Initially, perhaps the most confusing point is that where IP addresses are 4 bytes, an OSI address can vary from 8 to 20 bytes. Additionally, whereas one may be comfortable with the dotted decimal format of an IPv4 address, ISO addressing may contain unfamiliar hexadecimal notation. An ISO address consists of three basic parts:

✓ The Area ID
✓ The System ID
✓ The NET Selector (NSEL)

49.0001	1270.1012.3123	00
Area ID	System ID	NSEL
1-13 bytes	6 bytes	1 byte

Table 6.3 NSAP Address Fields

The *Area ID* is the portion of the address that is variable, ranging from 1 to 13 bytes. Additionally, the Area ID itself has two portions. The first byte tells the system how to interpret the rest of the Area ID and is called the *Address Family Identifier* (AFI). The Area ID is an AS local trait. Most often, you will see the AFI byte set to 49, which was specifically reserved for private addressing. It is the same thing as using RFC 1918 IP addressing or private BGP AS numbers. The JNCIA exam will most likely use AFI 49 in examples.

The next 0 thru 12 bytes are the *domain identifier*. It is possible to have an Area ID comprised entirely of the AFI with no bytes allocated for a Domain ID. In such a case the entire NSAP would be 8 bytes long. If a 12 byte full-length Domain ID was used, the address would increase to 20 bytes. Allocation of these is up to the discretion AS administrator. Sample domain identifiers could include:

0007	two-byte hexadecimal
4311	two-byte hexadecimal
43ac	two-byte hexadecimal
0007.43ac	four-byte hex
123c.ffab	four-byte hex

Following the variable Area ID, there is a *System ID* consisting of a fixed six bytes. These six bytes must be unique throughout the given area. Common practice is to encode the IPv4 Router ID into the System ID by converting the four byte IP address to six bytes. The simplest way to do this conversion is to keep the IP address in its decimal notation and fill out the octets with any necessary leading zeros. This means that each byte in the decimal notation has three digits. Therefore 192.168.123.45 becomes 192.168.123.045. The last octet becomes 045. The decimal notation is normally moved to cluster two byte hex groups, namely every four digits. Similarly:

10.1.12.34 becomes 010.001.012.034 is grouped 0100.0101.2034
10.10.101.10 becomes 010.010.101.010 is grouped 0100.1010.1010

As you can see, this is not really a mathematical conversion to 6 bytes. Rather it is just a simple way to fill out the required byte length while more or less keeping the integrity of a related IP address.

The last 1 byte is the *NET Selector*. For purposes of the JNCIA, the NET on IS-IS routers is always set to 00. When set to 00, the NSEL indicates we are dealing with a router NSAP. When dissecting NSAP addresses, it may be helpful to begin at the right hand side. We will see why in a moment. Looking at a sample ISO address:

49.0101.2421.2112.00

Beginning at the right, the final byte is the NSEL of 00 identifying this as a router. The next section is made up of the IP address 10.124.212.112 converted to six bytes 0101.2421.2112. The first byte is the AFI of 49. There are no additional bytes allocated for a domain identifier. Consequently, the entire Area ID is one byte.

If we look at a slightly more complex sample:

49.43ac.0730.0001.1921.6804.2129.00

Beginning at the right, we identify the router NSEL of 00. Next, we expect a six byte System ID, and can see it is the IP address 192.168.42.129. Now we have a large section of hexadecimal notation. Consequently, the entire Area ID is seven bytes. Keep in mind that the only thing of variable length is the domain identifier portion of the Area ID. The AFI is mandatory, and is only one byte. Therefore the next six bytes, 43ac.0730.0001, are the Domain ID. This leaves the last byte to be the AFI of 49.

One last note on ISO addressing is dynamic mapping of ISO *System Identifiers* (sysids). JUNOS supports the configuration of a host name for each system. Once this is established, the router can carry this name with a sysid in a dynamic hostname *type length value* (TLV) in LSP packets that are transmitted.

IS-IS Packets

IS-IS communicates with packets called *protocol data units* (PDUs) exchanged between active routers. There are four different types of packets used to exchange information:

- *IS-IS Hello* (IIH) PDU – Broadcast to determine neighboring IS-IS systems. Discovers and differentiates between Level 1 and Level 2 routers.

- *Link State PDU* (LSP) – Contains information regarding the state of the connections to adjacent routers. Periodically, LSPs are flooded throughout the IS-IS area.

- *Complete Sequence Number PDU* (CSNP) - Contains an entire listing of LSPs in the database. CSNPs are sent out all IS-IS links periodically to ensure all routers have synchronized LSP databases. A designated router (the DIS) will multicast the CSNP on a broadcast network media to limit acknowledgement traffic.

- *Partial Sequence Number PDU* (PSNP) – Request for update PDU. When a CSNP receiving router detects a fault with its LSP table (when it is out of date or not synchronized), it sends a PSNP to the router that originally transmitted the CSNP requesting the missing LSP. The PNSP transmitting node is then forwarded the missing LSP.

Details on these communication packets are also listed in the diagram below.

PDU	Format	Target	Purpose/Function
IS-IS Hello (IIH)	Broadcast	Neighboring systems	Discover neighbors and determine neighbor level(s)
Link-state (LSP)	Multicast	All routers in the particular level	Announces state of adjacencies to neighboring systems
Complete Sequence Number (CSNP)	Flooded (multicast by DIS)	All routers in an area	Contains complete database to ensure all routers are synchronized
Partial Sequence Number (PSNP)	Multicast	CSNP transmitter	Sent by a receiver to the transmitter to indicate a missing LSP. The transmitter then forwards the link to the receiver who sent the PSNP

Table 6.4 IS-IS PDU Descriptions

Communication between IS-IS devices

There are three types of hello packets: LAN Level 1, LAN Level 2, and Point-to-Point. LAN traffic is sent to a specific MAC address that specifies all routers in that particular area. Level 1 packets are sent to the MAC address of 01-80-C2-00-00-14 (signifying "all Level 1 IS-IS routers") and Level 2 packets are sent to 01-80-C2-00-00-15 (signifying "all Level 2 IS-IS routers"). Each different type of packet is tagged with a different PDU type to identify the purpose of the message. IS-IS uses sequence numbers and aging to ensure routers have a full and updated routing database. It is important to know the PDU type for debugging purposes, but this information is not pertinent to the JNCIA exam.

The *hold time* states how much time can pass with no communication before a router is declared dead. The *hello-interval* states how often to send Hello packets to neighbors. The hold time and hello-interval are set by default to 27 and 9 seconds, respectively, within JUNOS. Both of these values can be adjusted under either or both levels of any interface. When communicating with the DIS router, the timers configured are divided by three (making the default hold time and hello interval for the DIS 9 and 3 seconds, respectively). Unlike OSPF, the hold timer is reset back to its maximum value when ANY message is received from that neighbor (not just a Hello message).

JUNOS supports simple and MD5 authentication on IS-IS interfaces. For routers to communicate on authenticated interfaces, the information must match for the domain, area, and all interfaces in question. Interfaces can be configured to generate authenticated packets, but simultaneously not check authentication on received packets using the `no-authentication-check` configuration option.

IS-IS Configuration

The top level of IS-IS is at the `[edit protocols isis]` section of the configuration tree. The minimum IS-IS configuration within JUNOS consists of three steps:

1) IS-IS must be enabled globally (specifying which interfaces to run on).
2) A network entity title (NET) must be configured on an interface (usually the NET is configured on a loopback interface).
3) Each IS-IS interface must have family ISO configured.

To configure a NET on unit 0 of loopback0, issue the following command:

```
set interface lo0 unit 0 family iso address
49.0000.0000.0001.00 passive
```

To specify which interface will run IS-IS, issue the following command:

```
set protocols isis interface so-1/0/0.7
```

To configure family ISO on an interface, issue the following command:

```
set interface so-1/0/0 unit 7 family iso
```

The commands issued above will yield the following configuration on the router:

```
[edit protocols]
jncia@my.router1# show
       isis {
             interface so-1/0/0.7;
             interface lo0.0 {
                 passive;
             }
         }
```

Interface configuration:

```
[edit interfaces]
jncia@my.router1# show
     so-1/0/0 {
         unit 7 {
             family iso;
         }

     lo0 {
         unit 0 {
             family iso {
                 address 49.0000.0000.0001.00;
             }
         }
```

Recall that a router requires only a single NSAP address to participate in IS-IS. This NET identifies the router itself; individual interface NSAPs are not required.

Additionally, optional parameters are available under the [protocols isis] level of configuration mode. As noted previously, an IS-IS interface can be configured to participate in IS-IS level 1, IS-IS level 2 or both levels. By default, an interface running IS-IS belongs to both level 1 and level 2. The IS-IS level of an interface is configured in the global IS-IS configuration section. Note that when configured for both levels, the 'L' tag listed under 'show isis interface' will be '3'.

For an IS-IS interface to belong to a single level, the undesired level must be disabled. To disable levels on an IS-IS interface, issue the following command:

```
[edit protocols isis]
jncia@my.router1#  set  interface  interface-name
level [1|2] disable
```

Any IS-IS enabled interface defaults to a link metric of 10. To change this value the following command statement is used:

```
[edit protocols isis]
jncia@my.router1#  set  interface  interface-name
level [1|2] metric metric
```

Where `metric` is the desired link cost and `level-1|level-2` indicates the link type.

Authentication may be enabled on all IS-IS levels (at the global level as illustrated below), per level, per interface, and per level on an interface.

```
[edit protocols isis]
jncia@my.router1# set authentication-type type

[edit protocols isis]
jncia@my.router1# set authentication-key key

[edit protocols isis]
```

```
jncia@my.router1#   set   interface   interface-name
level [1|2] authentication-type type
```

Where the authentication-type may be md5 or simple and the key may be an ASCII string that will either be encrypted or plaintext depending on the authentication type. Remember that JUNOS requires password strings to be enclosed in quotes if they include a space. A router may also be configured to generate authenticated packets, but ignore authentication on received packets. This is done in the following manner:

```
[edit protocols isis]
jncia@my.router1# no-authentication-check
```

Mesh Groups

Routers connected in a full mesh will waste a lot of time and bandwidth sending each other LSP packets. To reduce some of this communication, routers can have *mesh groups* configured on particular interfaces. Interfaces can then be told to stop the flooding of LSPs through that interface by issuing the set mesh-group blocked command.

```
[edit protocols isis interface interface-name]
jncia@my.router1# set mesh-group [blocked|value]
```

IS-IS Traffic Engineering

IS-IS traffic engineering extensions are *Type Length Values* (TLVs) that describe link attributes. TLVs are included in IS-IS link state PDUs and are used to populate the *Traffic Engineering Database* (TED). The information in the TED is used by CSPF to compute paths for RSVP signaled MPLS LSPs. IS-IS does not use MPLS LSPs as next hops, by default. For IS-IS to use LSPs, IS-IS traffic engineering shortcuts must be enabled. To enable these shortcuts, issue the following command from edit mode:

```
set protocols isis traffic-engineering shortcuts
```

Monitoring IS-IS in JUNOS

The basic commands to assist in troubleshooting and maintaining IS-IS are listed below.

```
jncia@my.router> show isis ?
Possible completions:
  adjacency      Show the IS-IS adjacency database
  database       Show the IS-IS link-state database
  hostname       Show IS-IS hostname database
  interface      Show IS-IS interface information
  route          Show the IS-IS routing table
  spf            Show SPF calculation information
  statistics     Show IS-IS performance statistics
```

```
show isis adjacency <system-id> <brief|detail|extensive>
```

This command indicates the status of each adjacency on the router. The system-id and brief parameters are optional.

```
jncia@my.router> show isis adjacency brief
IS-IS adjacency database:
Interface    System    L State  Hold     SNPA
fe-0/0/0.2   Denver1   1 Up     8    0:60:94:a3:73:dd
fe-0/0/0.3   NewYork   2 Init   25   0:20:35:e7:3d:53
fe-0/0/0.3   NewYork   1 New    15   0:20:35:e7:3d:53
```

From the ‘L’ column we can distinguish what IS-IS level is functioning on that interface. The ‘State’ column indicates whether or not the adjacency is up. The ‘System’ column will return the hostname of the remote node because of IS-IS dynamic sysid mapping. In this case, the adjacency to the `Denver1` router is functioning, however the two connections to `NewYork` are in the process of coming up.

```
show isis interface <interface-name> <brief|detail>
```

This indicates the status of each IS-IS interface on the router. The specific interface name and brief parameters are optional.

```
jncia@my.router> show isis interface
IS-IS interface database:
```

```
Interface        L CirID Level 1 DR      Level 2 DR
fe-0/0/0.2       1   0x2 Denver1         Disabled
fe-0/0/0.3       3   0x3 my.router       my.router
lo0.0            0   0x1 Passive         Passive
```

As with 'show isis adjacency', the 'L' column will tell us what level this box belongs to. However, in this case, a value of 3 indicates the router is an L1/L2 device. Similarly, a value of 0 indicates a passive interface. The final two columns will display the active DR for the interface. `Fe-0/0/0.2` reports the neighbor `Denver1` is the L1 DR. There is no L2 DR for that segment because the interface is L1 enabled only. Interface `fe-0/0/0.3` reports that this router, `my.router`, is both the L1 and L2 DR.

`show isis database <system-id> <brief|detail>` outputs the contents of the ISIS link-state database assembled from compiled PDUs. Similar to how one can display the contents of the OSPF database, a user can view the ISIS table in its entirety or for a specific system.

```
jncia@my.router> show isis database
IS-IS level 1 link-state database:
LSP ID                 Sequence Checksum Life(sec)
Denver1.00-00              0x30  0x932f    704
Denver1.02-00              0x4   0x4d5c    1184
my.router.00-00            0x27  0xb55e    1159
my.router.03-00            0xd   0         0
NewYork.00-00              0x23  0xcd32    1130
  5 LSPs

IS-IS level 1 link-state database:
LSP ID                Sequence   Checksum Life(sec)
Denver1.00-00         0x1f       0               0
my.router.00-00       0x25       0x79f2       1166
NewYork.03-00         0xf        0               0
NewYork.00-00         0x1f       0xf60d       1123
  4 LSPs
```

```
show isis statistics
```

This command will display the PDU performance statistics for an active IS-IS router. This includes the number of transmitted, received, processed, dropped, and retransmitted PDUs. Such information can be helpful when troubleshooting bad connections and circuits.

```
jncia@my.router> show isis statistics
IS-IS statistics for my.router:

PDU type     Rcv’d  Proc’d     Drops     Sent     Rexmit
LSP             64      64         0       123         0
IIH           3164     655      2509      3140         0
CSNP           172     172         0        50         0
PSNP            23      23         0         0         0
Unknown          0       0         0         0         0
Totals        3423     914      2509      3313         0

Total packets received: 3423 Sent: 3313

SNP queue length:           0 Drops:          0
LSP queue length:           0 Drops:          0

SPF runs:                 125
Fragments rebuilt:        172
LSP regenerations:         57
Purges initiated:          11
```

Below is the output from the ISIS logfile with the ‘state’ traceoption flag set illustrating the steps undertaken while an adjacency forms.

```
Adjacency state change, my.lab1, state Up->Down
interface fxp0.3, level 2
Adjacency state change, my.lab1, state Down->New
interface fxp0.3, level 2
Adjacency state change, my3.lab1, state New->Init
interface fxp0.3, level 2
Adjacency state change, my3.lab1, state Init->Up
interface fxp0.3, level 2
```

Key Points

- IS-IS is a link-state Interior Gateway Protocol.
    - Link state protocols require that each participating node have full knowledge of the complete network topology. Each router must keep track of the link state of each of its connections and immediately notify the other nodes of any changes occurring.
- Utilizes the Djikstra algorithm to run SPF calculations.
- Configured with OSI NSAP addresses that range from 8 to 20 bytes and contain:
    - A variable length Area ID
    - A six-byte System ID
    - A single byte NET Selector (NSEL)
- The NSAP Area ID normally includes a single byte AFI of 49.
- IS-IS is hierarchical, the domain can be broken into two logical area types.
    - Level 1 routers – Route within an area and to a Level 2 router when routing to a destination outside the area
    - Level 2 routers – Route packets between Level 1 areas and to other ASs.
- Multi-access networks utilize a DIS to reduce route update overhead
- Interfaces configured to run IS-IS are entered into both level 1 and level 2, by default. Either or both of these levels can be disabled on a per-interface basis.
- IS-IS uses ISO addressing for node identification rather than IP.
    - The ISO addressing consists of an Area ID, system ID, and NSEL
    - The AFI is the first byte of the Area ID
- Four types of protocol data units (PDUs):
    - IS-IS Hello (IIH)
    - Link State PDU (LSP)
    - Complete Sequence Number PDU (CSNP)
    - Partial Sequence Number PDU (PSNP)

Additional Information (RFCs)

More information can be found within:

- ISO/IEC 10589
- RFC 1195 Use of OSI IS-IS for Routing in TCP/IP and Dual Environments

- RFC 1237 Guidelines for OSI NSAP Allocation on the Internet
- RFC 3277 IS-IS Transient Blackhole Avoidance

Chapter Seven

BGP

Chapter 7: BGP

Overview

This chapter will cover Border Gateway Protocol (BGP). The topics covered will include the protocol's evolution, implementation, features, and configuration within JUNOS. BGP is fairly simple to configure and operate at a basic level. However, as can be seen by the large number of RFCs, the protocol has a multitude of options and is still under development. By the end of this chapter you should understand and be able to define:

- ✓ The purpose of using BGP in a wide-area routing environment.
- ✓ Internal vs. External BGP.
- ✓ BGP neighbor configuration.
- ✓ Messages exchanged between BGP neighbors, or peers.
- ✓ BGP route selection.
- ✓ Route reflection within a BGP mesh.
- ✓ Basic diagnostic commands for BGP within JUNOS.

Introduction

BGP is considered an advanced distance-vector protocol. It communicates on TCP port 179 and establishes explicit neighbor adjacencies with which it can exchange detailed routing information. BGP is the protocol used to exchange routes at the core of the Internet between service providers. It is also widely used to connect customers to these providers.

Up to this point, we have reviewed internal gateway protocols (IGP) such as RIP, OSPF, and IS-IS. These protocols are used for the exchange of routing information within a single administrative domain. However, to properly connect external networks (those under alternate administrative control), an exterior gateway protocol (EGP) is required. BGP allows this external routing exchange by recognizing the boundary of these domains and assigning them an identifier known as Autonomous System number.

Service providers are assigned one or more Autonomous System (AS) numbers that are used to identify themselves (in the scope of BGP) to other BGP speakers. An AS number is 16-bits, ranging of 1 to 65535. AS designators throughout the designation 1-64511 are assigned to the public, while 64512-65535 are considered private space (much like RFC 1918 private IP address space).

BGP resembles RIP in that one of the primary methods of route selection is based upon hop count. The key difference being that BGP considers one AS to be a 'hop'. BGP routes keep record of the Autonomous Systems that they transit to reach their destination. This feature is referred to '*as-path*' length.

Full routing tables are exchanged when routers first form an adjacency. This default behavior can be easily altered through the use of policy. From that point forward, only incremental changes are sent as routes are added, removed, or their attributes are changed. This limits the amount of protocol overhead needed once a session is established.

Versions

Every modern ISP today runs BGP version 4 to interconnect with other service providers. The only key change in BGP version 4 (with respect to the JNCIA exam) is the addition of classless inter-domain routing (CIDR). Version 4 allows the ability to remove classes from routing and summarize addressing to reduce the overall number of routes advertised.

External BGP vs. Internal BGP

BGP can be logically separated into two forms, internal and external. Internal BGP (iBGP) indicates a connection with a BGP peer in the same AS, while external bgp (eBGP) peers are those in another AS. It is important to distinguish between the two, as some of the key rules of BGP are applied to each.

BGP Basic Rules and Terms

There a just a handful of "golden rules" to remember about BGP:

- When a router receives a route from a BGP peer, it checks for its own AS. If it is in the route anywhere, the router discards the route to avoid a loop.
- When a router receives an eBGP route (one not from its own AS), it prepends its own AS to the front of the route's *AS-path* before sending it to other peers. The exception is when the route is sent to

internal neighbors (if internal neighbors saw their own AS in the path, they would drop the route advertisement to avoid a routing loop).

- eBGP peers will modify external advertisements to make themselves the next-hop (thus making the next-hop reachable by all internal peers). By default, iBGP peers do not modify the next hop attribute of a received route.
- A route learned via an eBGP peer will be sent to all of that router's BGP peers.
- A route learned via iBGP will be advertised to connected eBGP neighbors.
- A route learned via iBGP will not be re-advertised to any other iBGP peer. This is also sometimes known as "BGP Split Horizon". This effectively limits iBGP propagation to immediately adjacent peers, and thus requires a full-mesh for an iBGP network to ensure all routers receive all routes and updates. The use of route reflectors or confederations averts this rule (See the below section on route reflection).
- iBGP peering connections do not use the multi-hop command (explained below), but simply follow the rule that the peer's address must be reachable via the internal routing protocol (Internal Gateway Protocol, or IGP).

Message types

BGP messages range from 19 to 4096 bytes and consist of four message types:

Open, Update, Notification, and *Keepalive*.

Open message:

Begins the BGP peering session communication. After the TCP connection comes up, either router can send an Open message. This is a unicast message sent to the IP address configured for the peer on TCP port 179. It will contain:

- Version (currently will be version 4).
- Autonomous System (AS) number.
- Hold-time values.

- BGP identifier (IP address of sending router, also known as router ID).
- Optional parameters tag (zero in this field indicates no optional parameters).
- Optional parameters.

Update messages:

Contain all route prefixes and their attributes. Update messages are sent to add, withdraw, and change the attributes of BGP routes.

- Once a peering session is established, update messages are sent to inform the peer of all routes configured for advertisement.
- When a route's attributes change, JUNOS will automatically send an update message to BGP peers to notify them of the changes that have been made.

Notification messages:

Used to terminate a BGP peering session gracefully.

- Triggered if a router experiences a failure that necessitates the closure of the BGP session.
- Sent during the exchange of Open messages if a problem is encountered (wrong version number, bad peer AS, etc.).

Keepalive messages:

Necessary to ensure BGP sessions remain up.

- Value of timers is negotiated during the exchange of Open messages. If timers are changed on an existing session, the session will drop and re-establish with the new parameters.
- Default *holdtime timer* for JUNOS is 90 seconds (time allowed with no Keepalive or update messages before the session is terminated).
- *Keepalive timer* is 1/3 of the *holdtime* (default=30 seconds).
    - If you increase or decrease the holdtime, the Keepalive interval will be altered accordingly to maintain the 1:3 ratio

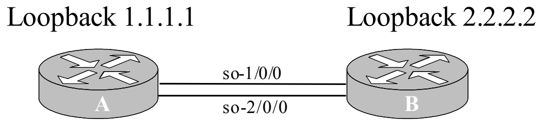
BGP Route selection

After ensuring that the Next-hop address is reachable, if more than one route exists to the target destination, the route to install as active will be chosen by the following method. This process also holds true for multiple routes to the same neighbor.

1. Highest ***Local Preference***
   a. Value of 0-4294967295, default is 100.
   b. Used to manipulate how traffic will leave the AS.
   c. This value is passed to iBGP peers only, not eBGP peers.
2. Shortest ***AS-Path***
   a. Each AS number in the as-path string is given a value of 1.
   b. The AS-Path is incremented by at least one hop and sent to both iBGP and eBGP peers (however the local AS path is not added to iBGP advertisements).
   c. A shorter as-path length implicitly denotes a shorter path.
3. Lowest ***Origin Code***
   a. Internal (I) is lower than External (E), which is lower than Incomplete (?). Prefer internal paths, which include IGP routes and locally generated routes (static, direct, local, etc.).
   b. Note that JUNOS sets the code to internal when distributing routes from one protocol into BGP (most other vendors set these to '?')
4. Lowest ***Multi-Exit Discriminator (MED)***
   a. Default value is 0. No MED value is interpreted as 0.
   b. By default, MED value is only compared if the closest AS (the peer AS the route was directly learned from) is the same. However, the command "*path-selection always-compare-med*" can be used to force JUNOS to compare the MED value regardless of the AS the route was learned from
5. Prefer ***eBGP paths over iBGP paths***
6. ***Lowest IGP cost***, or metric (Closest next-hop address)
7. Prefer paths from the ***higher routing table number*** (inet.3 is preferred to inet.0)
8. Lowest ***router ID*** (or loopback IP address, if no router ID is specified) of the BGP router advertising the route
9. Prefer paths with the ***largest number of next hops***.
10. Prefer the ***lowest peer IP address***

Normally, eBGP peers are configured to the physical peer IP address. The BGP route selection process does not inherently load balance over equal cost paths to the same neighbor. This principal is brought to light at the path selection criteria number 8 above which is a tie-breaker that will result in a single path being chosen. However, there are situations in which multiple circuits are connected to one provider on the same router and load balancing is required over these paths. In these situations, *BGP multihop* is configured to provide the solution.

BGP multihop can be configured on external peers to allow for the peer to be more than one network interface away. Note that the router loopback is considered a logical network interface. Therefore, peering to a router's loopback address is considered to be a distance of **two** network interfaces (thus requiring *multihop*). When peering to loopback addresses, multiple physical links can be combined into one logical BGP session. This action, combined with the addition of the *multipath* setting is the only method for true load balancing when using BGP. When *multipath* is configured on a session, the Router ID (loopback) for both physical circuits is the same. Therefore, it is possible to utilize both paths.



By peering with the loopback address on each router and enabling multipath, BGP traffic from router A-B and B-A can load balance over both SONET links equally

Figure 7.1 BGP Multipath

By default, BGP will accept all routes imported from any protocol. However, it will advertise only BGP routes to its peers (ignoring those from other protocols). This default behavior can be altered through policy manipulation (discussed in the Policy chapter).

Route Damping

BGP allows for a feature called route damping. This feature allows routes that have been withdrawn and readvertised, or 'flapped', more than a certain number of times in an allotted time interval to be placed in 'holddown'. The timers and settings are completely configurable within JUNOS and will vary by application. *Route Damping* is not active by default in JUNOS. *Route Damping* is key to the stability of the Internet, but it is not covered in depth for the JNCIA.

BGP Finite-State Machine

The BGP neighbor adjacency process is commonly referred to as the 'Finite-State Machine', which implies that there are specific input variables that will trigger a finite number of possibilities.

There are 13 BGP events that are the trigger for, or the result of, a state change. Details of these events are important to know, but they are not covered in detail for the JNCIA exam. These are the BGP events defined in the most recent RFC:

- BGP start
- BGP stop
- BGP transport connection open
- BGP transport connection closed
- BGP transport connection open failed
- BGP transport fatal error
- Connect-retry timer expired
- Keepalive timer expired
- Receive Open message
- Receive Update message
- Receive Keepalive message
- Receive Notification message

BGP session states

The states that a BGP peering session can be in are listed below. Following that is a diagram that may prove useful in recognizing the BGP connection process.

The below 3 states occur **before** the TCP session on port 179 is up:

Idle

The router is configured for BGP, but no action is being taken and all incoming sessions from this peer are refused. The router is awaiting a BGP start event to initiate the TCP session with the peer.

Connect

The router is waiting for the TCP connection to be completed. Once the TCP session is up, the router will send an Open message and move to Open-sent state.

Active

The TCP connection has failed in the Connect state and the router is continuing to listen for a TCP connection from the peer. If the TCP session is now down, the router will back to Connect. If the TCP session is still alive, the router will send another Open message and move back to Open-sent status.

Below states occur **after** the TCP session on port 179 is up:

Open-sent

The router has completed its work and has sent an Open message. It is waiting for an Open message from the peer. Upon receipt of this message from the peer, if no errors are detected, the route will send its first Keepalive message and move to Open-confirm.

Open-confirm

The router has sent its first Keepalive message and is waiting for a Keepalive message from the peer.

Established

A Keepalive message is received from the peer and the session is up. Routing updates will now be exchanged.

It is important to note two things about BGP state for the JNCIA exam.

1. ‘Active’ does not indicate a working BGP session (the term can sometimes be misleading).
2. It is rare that you will see a session marked as “Established”, but rather three numbers will appear in the far right column indicating the number of routes from that peer that have been received, are

active, and are dampened. Output from a sample BGP peering session is inserted below the BGP state diagram.

Below is the output from the BGP log file with the ‘state’ traceoption flag set illustrating the steps undertaken while an adjacency forms.

```
jncia@my.router1> show log bgp
bgp_event: peer 10.2.2.1 (Internal AS 11) old state Idle
event Start new state Active
bgp_pp_recv: dropping 10.2.2.1 (Internal AS 11),
connection collision prefers 10.2.2.1+2287 (proto)
bgp_peer_close: closing peer 10.2.2.1 (Internal AS 11),
state is 3 (Active)
bgp_event: peer 10.2.2.1 (Internal AS 11) old state
Active event Stop new state Idle
bgp_event: peer 10.2.2.1 (Internal AS 11) old state Idle
event Start new state Active
bgp_event: peer 10.2.2.1 (Internal AS 11) old state
Active event Open new state OpenSent
bgp_event: peer 10.2.2.1 (Internal AS 11) old state
OpenSent event RecvOpen new state OpenConfirm
bgp_event: peer 10.2.2.1 (Internal AS 11) old state
OpenConfirm event RecvKeepAlive new state Established
```

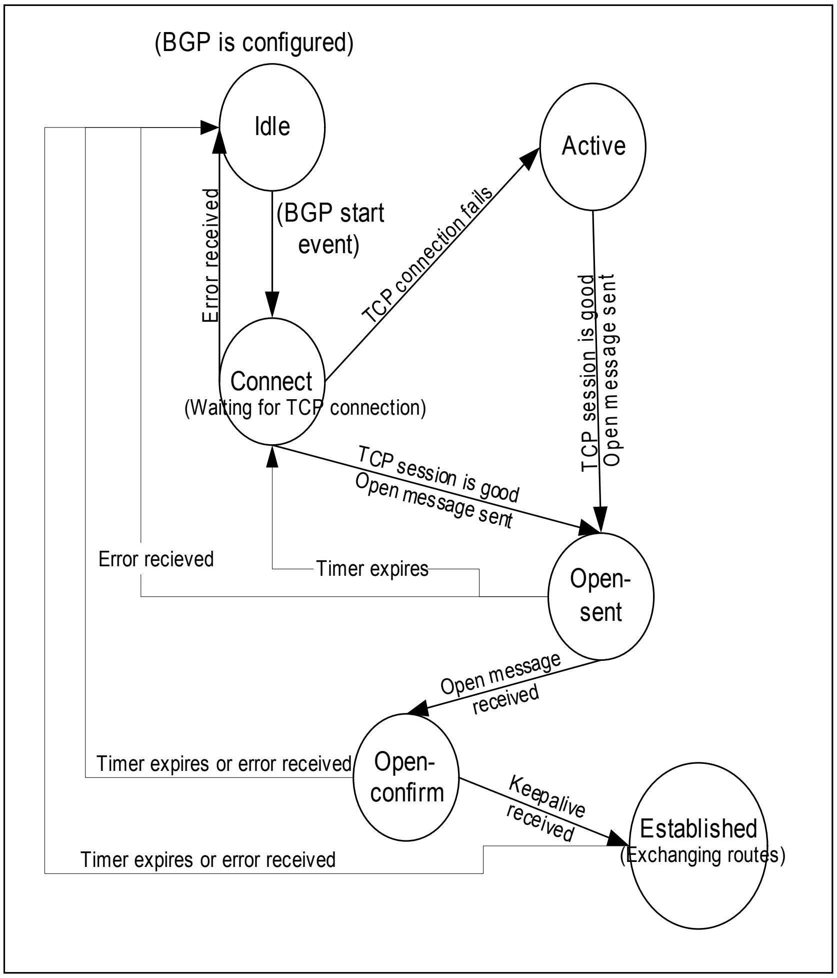


Figure 7.2 BGP connection process

On the next page, you'll see a sample BGP neighbor view:

```
jncia@my.router> show bgp summary
Groups: 9 Peers: 10 Down peers: 1
Table          Tot Paths   Act Paths  Suppressed  History  Damp State    Pending
inet.0            245788      122217           0        0           0          0
inet.2                 0           0           0        0           0          0
Peer            AS     InPkt   OutPkt   OutQ    Flaps  Last Up/Dwn State|#Active/Rcvd/Damped
10.244.2.145    65333   1076    919        0        3    5d16h  113354/121636/0      0/0/0
10.244.2.154    65333    452    432        0        4    5d19h  8250/121636/0        0/0/0
10.246.46.126   65333    300    355        0       47    4d13h  25/40/0              0/0/0
10.210.41.166   11315    640    641        0       42    3d15h  10/23/0              0/0/0
10.210.62.58    11315    651    671        0       41    3d15h  15/23/0              0/0/0
10.158.57.54    13564    779    887        0        4    3d15h  6/10/0               0/0/0
10.101.246.58   12345      0      0        0        0    4d7h   Active
```

Route reflection

We have discussed that all routers in the iBGP network must be fully meshed in order to work properly. A full mesh requires a large number of connections and will slow convergence while consuming a great deal of bandwidth. A network with twenty routers, for instance, would require each node to support and converge nineteen separate BGP sessions.

To eliminate this scalability issue, *route reflection* is one of the two methods deployed in almost every ISP network (confederations being the second). Route reflection is simple to configure and allows BGP to break some of the "Golden Rules" in order to reduce the number of top-level connections. It is commonly deployed on a per-POP basis, allowing just one or two routers in each physical location to be part of the top level iBGP mesh. The rest of the routers in that site are the clients and do not exchange routes with other physical locations.

BGP route reflection is most easily viewed in a hierarchical manner. Some of the network routers are transformed to route reflector clients and simply placed in a 'lower tier' behind their route reflector server. **All** BGP updates, either internal or external, will be passed from the server to its client(s). This means that they rely on their servers for all BGP updates. All client updates are sent **only** to their respective server for propagation to the rest of the network.

A group of clients and their respective servers is referred to as a cluster. All routers in the cluster will share the same Cluster ID.

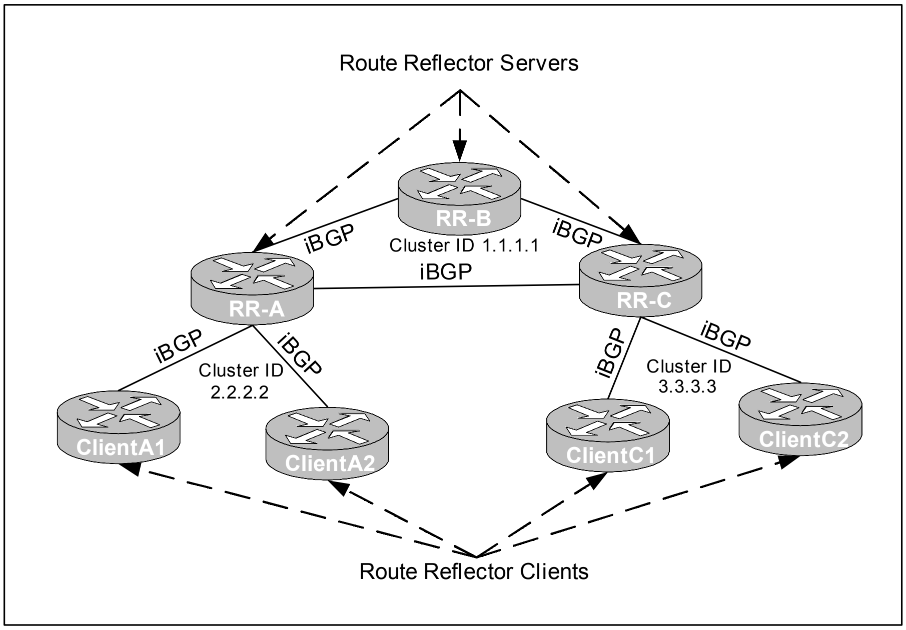


Figure 7.3 Simple BGP Route-Reflection

Configuration on a client is no different than any other BGP session. The client just thinks it is part of a very small BGP mesh (only one or two peers). It is not aware that those connections are actually to its servers, which will feed it all of its non-local BGP routes.

Configuration on a BGP route reflection server takes simply the addition of a *Cluster ID* on the server. This is an arbitrary IP address (although it must be unique to each cluster) that helps prevent routing loops between route reflectors by allowing them to ignore routes reflected to them from iBGP peers within the same *Cluster*.

Confederations

Confederations represent the second way to overcome the full-mesh problem associated with BGP for internal networks. They allow the ability to divide the AS into multiple internal Confederations identified to one-another with the use of the *Member-AS* number. This number (often private AS numbers 64512-65535 are used) is removed from advertisements before they leave the primary AS so that confederations are not visible to external peers.

Confederations are not covered in detail on the JNCIA exam.

BGP Configuration within JUNOS

BGP is first activated on the router, and then each peer is configured. Every BGP peer must be exclusively defined before forming an adjacency, and thus exchanging routes.

Before activating BGP neighbors, the local autonomous system ID must be defined at the global level of "edit routing-options". In addition, the *router-id* may be defined. If this attribute is not defined, the IP address of the first router interface is used as that router's ID.

The top level of configuration for BGP is under 'edit protocols bgp'. At this level, global BGP options can be set and groups can be defined. Each group must contain at least the following information:

```
[edit protocols bgp]
group group-name {
      peer-as number
      type [external | internal]
      neighbor address
      }
```

To allow for better scalability, it is desired to group BGP peers based upon common attributes (peer AS, external, internal, etc.) and then define more specific information (neighbor address, etc.) under each individual neighbor within the group.

Multiple values can be defined globally, under each BGP group, and again under each peer (except export policy, as noted below).

`[edit protocols bgp]` (global) or
`[edit protocols bgp group group-name]` (group) or
`[edit protocols bgp group group-name neighbor address]` (peer)

Remember that more specifically assigned values override less-specifically assigned (neighbor overrides group which overrides global).

```
authentication-key key;
```

JUNOS supports MD5 authentication on BGP sessions; it is disabled by default.

`cluster` *`cluster-identifier`*

Used to identify a route reflector.

`description` *`"name";`*

`export [`*`policy-names`*`];`

NOTE: Export policy only on group or global level; individual peers will use their group's export policy.

Export policy or policies are applied to routes from the routing table before they are advertised to the group or peer.

`hold-time` *`seconds;`*

Seconds allowed to pass without hearing a Keepalive before the session is declared dead, default for JUNOS is 90 seconds.

`import [`*`policy-names`*`];`

Import policy or policies are applied to incoming advertisements before they enter the routing table.

`local-address` *`address;`*

Used if a specific address is desired to be used on this session or sessions, other than the default.

`local-preference` *`value;`*

Applied to routes learned from this peer or neighbor only at time of advertisement to internal peers

`metric-out (`*`metric`*` | minimum IGP <`*`offset`*`> | igp <`*`offset`*`>)`

Sets a metric for outbound routes using the MED attribute.

`metric`: A value from 0-4,294,967,295 (by default no metric sent)

`minimum IGP`: Set the metric to the minimum IGP cost to reach this prefix. If a new metric comes along that is lower, change to the new, lower value (never raise). This can be offset by any value from (-2^{31}) to (2^{31}-1).

`IGP`: Set MED value to the most recent IGP cost to this prefix.

`Offset`: Can offset any value from (-2^{31}) to (2^{31}-1).

`multihop <ttl-value>`

eBGP sets the TTL of TCP packets to one by default; if the multihop tag is entered, the default value is set to 64 (and can be set to anything from 2-255). This command is needed if using any address other than the directly connected address on the interface (even the loopback of the router counts as another hop). The purpose of using multihop on eBGP sessions is to provide link redundancy or load balancing. If two links are used to connect two routers, the common configuration will peer each router to the other's loopback address. When both links are up, packets will be sent across both circuits to the BGP peer (assuming no further configuration that would prefer one connection). If one link fails, the neighbor's loopback address can still be reached via the second connection.

`multipath`

Allows multiples paths to be installed in the forwarding table for a single prefix, thus enabling load sharing

`passive`

When used, this peer or group will not send Open messages, but will respond to Open messages received and subsequently form adjacency, if all other configurations are proper

`peer-as autonomous-system`

Specifies Peer AS. This is confirmed upon receipt of an Open message from the peer. If the values do not match, the session will not establish.

`preference value;`

Sets the local-preference value of BGP routes learned from this group or peer (range 0 to 4294967295 [2^{31}-1]; default 170)

```
prefix-limit {
           maximum number;
           teardown percentage idle-timeout
           [forever | minutes];
      }
```

This specifies the maximum number of prefixes that will be allowed from the peer or group. Once the number is reached, a log message will be generated. If the teardown statement is added with a percentage, messages are generated when the percentage is reached and the session is taken down once the limit is reached. It will be re-established rapidly unless a idle-timeout timer is specified to keep the session down for that amount of time before setting up again.

```
traceoptions {
           file name replace <size size> <files #>
      <no-stamp> <world-readable | no-world-
      readable>;
           flag flag <flag-modifier> <disable>;
      }
```

Traceoptions allow system log files to be created that monitor events, as configured; they will be discussed in detail in the JUNOS chapter.

```
type [internal | external];
```

Indicates iBGP or eBGP session.

Monitoring BGP within JUNOS

The helpful commands to perform basic BGP troubleshooting on a Juniper M-series router are listed below with some examples and explanations.

```
jncia@my.router> show bgp ?
Possible completions:
  group       Show the BGP group database
  neighbor    Show the BGP neighbor database
  summary     Show overview of BGP information
```

```
show bgp summary
```

The command gives a brief display of all configured BGP sessions on the router.

```
jncia@my.router> show bgp summary
Groups: 1 Peers: 2 Down peers: 1
Table  TotPaths  ActPaths   Suppressed  History   Damp State   Pending
inet.0     0         0           0          0           0          0
Peer          AS    InPkt   OutPkt  OutQ   Flaps LastUp/Dwn State|#Act/Rcv'd/Damp…
10.44.2.20   65536   497       21     0   13 1d 15:02:13 1/1/0
10.5.100.2   65530   371       20     0   13 1d 14:12:10 1/1/0          1/1/0
10.44.2.22   65536    36        9     0   14 1d 15:04:10 Active
```

The most important items to recognize are the `Peer`, `AS`, and `State|#Act/Rcv'd/Damp…` columns. Peer and AS identify the remote side of the BGP session. By default, if the AS number if different than the local routers AS number, it is an eBGP session. If they are the same, it is an iBGP session. The final column indicates either the State of the neighbor or the number of Active/Received/Damped routes. Note that 10.44.2.22 is in ‘Active’ and lists no numbers for routes. If there is more than one column of numbers under this heading, the second set indicates multicast routes for MBGP.

```
show bgp neighbor <neighbor-IP>
```

This command will show more detailed information for individual peers.

```
jncia@my.router> show bgp neighbor 10.44.2.20
Peer:10.44.2.20 +179 AS 123 Local:10.44.2.10+1026 AS 456
 Type: Internal State: Established Flags: <>
 Last State: OpenConfirm Last Event: RecvKeepAlive
 Last Error: None
 Export:[eBGP-Export-policy] Import:[eBGP-Import-policy]
 Options: <Preference LocalAddress HoldTime LogUpDown
Refresh>
  Local Address: 10.44.2.10 Holdtime: 90 Preference: 170
  Number of flaps: 0
  Peer ID: 10.44.2.20 Local ID: 10.44.2.10 Active
Holdtime: 90
  Keepalive Interval: 30
  NLRI advertised by peer: inet-unicast
  NLRI for this session: inet-unicast
  Peer supports Refresh capability (2)
  Table inet.0 Bit: 10000
    RIB State: restart is complete
    Send state: in sync
    Active prefixes: 1
    Received prefixes: 1
    Suppressed due to damping: 0
  Last traffic (seconds): Received 1 Sent 12
Checked 12
  Input messages: Total 40914842 Updates 2000
Refreshes 0 Octets 683417048
  Output messages: Total 714008 Updates 0
Refreshes 0 Octets 13566178
  Output Queue[0]: 0
  Trace options: state
Trace file: /var/log/bgplog size 10576 files 1
```

If the neighbor IP address is left out from this command, all neighbors will be listed in order of IP address. This will show all details about BGP neighbors. Key information to note is session information (how long it has been up and how many times it has bounced), and import and export policy for the group for which this peer belongs.

It is important to remember that policy plays a crucial role in combination with BGP. This chart will help you remember the commands and where they are applied. Each command is subsequently discussed in detail.

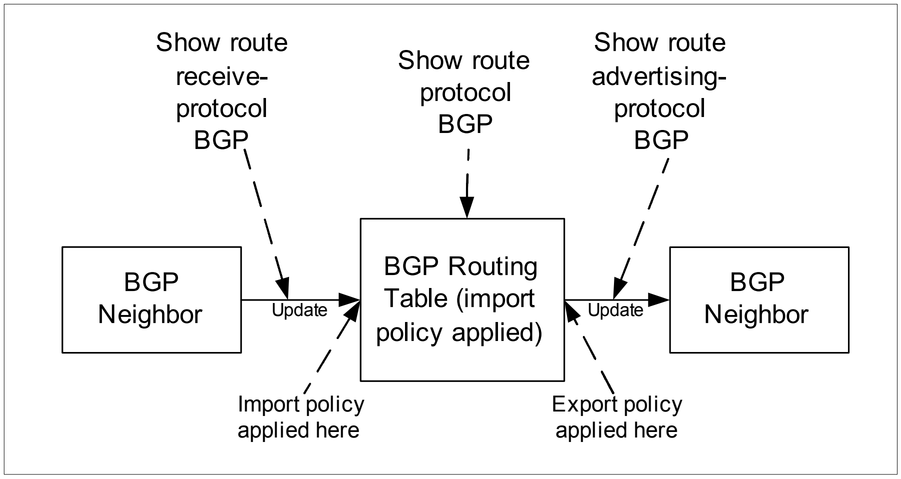

Show route receive-protocol BGP
Show route protocol BGP
Show route advertising-protocol BGP
BGP Neighbor
Update
BGP Routing Table (import policy applied)
Update
BGP Neighbor
Import policy applied here
Export policy applied here

Figure 7.4 Viewing protocol-specific routes before, with, and after policy is applied

```
show route receive-protocol bgp neighbor-IP
<destination>
```

List all received BGP routes from the specified peer (located in the adj-fib-in table). Output can be compared with the results from "show route protocol bgp *destination*" to determine the impact of the import policy on received routes.

```
jncia@my.router>show route receive-protocol bgp 10.2.2.1

inet.0: 20 destinations, 20 routes (20 active, 0
holddown, 8 hidden)
```

```
Prefix             Nexthop       MED   Lclpref AS
path
10.0.0.0/24        10.2.2.1                100 I
10.2.2.1/32        10.2.2.1                100 I
10.100.100.0/24    10.2.2.1                100 I
10.150.150.0/24    10.2.2.1                100 I
10.200.200.0/24    10.2.2.1                100 I
10.250.250.0/24    10.2.2.1                100 I
172.16.1.0/30      10.2.2.1                100 I
172.16.1.24/30     10.2.2.1                100 I
```

```
show route protocol bgp
```

List all active routes in the routing table that were learned from the protocol BGP.

```
jncia@my.router> show route protocol bgp

inet.0: 18 destinations, 18 routes (17 active, 0 holddown, 7
hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/24         [BGP/170] 00:32:25, localpref 100, from
10.2.2.1
                      AS path: I
                    > to 172.16.1.1 via fe-0/0/0.2
10.2.2.1/32         [BGP/170] 00:32:25, localpref 100, from
10.2.2.1
                      AS path: I
                    > to 172.16.1.1 via fe-0/0/0.2
10.150.150.0/24    *[BGP/170] 00:32:25, localpref 100, from
10.2.2.1
                      AS path: I
                    > to 172.16.1.1 via fe-0/0/0.2
10.200.200.0/24    *[BGP/170] 00:32:25, localpref 100, from
10.2.2.1
                      AS path: I
                    > to 172.16.1.1 via fe-0/0/0.2
10.250.250.0/24    *[BGP/170] 00:32:25, localpref 100, from
10.2.2.1
                      AS path: I
                    > to 172.16.1.1 via fe-0/0/0.2
172.16.1.0/30       [BGP/170] 00:32:25, localpref 100, from
10.2.2.1
                      AS path: I
                    > to 172.16.1.1 via fe-0/0/0.2
```

```
172.16.1.24/30      [BGP/170] 00:32:25, localpref 100, from
10.2.2.1
                      AS path: I
                    > to 172.16.1.1 via fe-0/0/0.2
```

```
show route advertising-protocol bgp neighbor-IP
```

List the BGP routes in the adj-fib-out table. This will list all BGP routes that have passed through the outbound policy filter and are advertised to the specific peer. This output can be compared to with the results from “show route protocol bgp *destination*” to determine the impact of the export policy on advertised routes.

```
jncia@my.router1>show route advertising-protocol
bgp 10.2.2.2

inet.0: 18 destinations, 18 routes (18 active, 0
holddown, 0 hidden)
Prefix             Nexthop       MED    Lclpref  AS
10.0.0.0/24        Self                     100   I
10.2.2.1/32        Self                     100   I
10.2.2.2/32        172.16.1.2    100        100   I
10.2.2.3/32        172.16.1.2    200        100   I
10.2.2.4/32        172.16.1.2    300        100   I
10.100.100.0/24    10.2.2.1                 100   I
10.150.150.0/24    10.2.2.1                 100   I
10.200.200.0/24    10.2.2.1                 100   I
10.250.250.0/24    10.2.2.1                 100   I
172.16.1.0/30      Self                     100   I
172.16.1.4/30      172.16.1.2    200        100   I
172.16.1.8/30      172.16.1.2    300        100   I
172.16.1.12/30     172.16.1.2    400        100   I
172.16.1.24/30     Self                     100   I
```

It is sometimes desired to view BGP routes before they are run through the import policy and after they are run through the export policy. With the use of these three commands, it is possible to tune BGP reception and advertisement to obtain the optimal routing scenario.

Because of the widespread use of BGP and its default behavior to accept all BGP routes and re-advertise all BGP routes, policy plays a large role in the manipulation of BGP. More details of policy and how it interacts with this protocol are covered in the Policy chapter.

Key Points and Summary

BGP plays a key role in the network of every Internet Service Provider. With the information provided in this chapter, you should be able to properly configure BGP peering adjacencies on Juniper routers. As noted in the introduction, there are many developments recently released and still in the works for this complex protocol. We did not cover protocol families which allow BGP to carry network layer reachability information (NLRI) for address families other than unicast IPv4 or IP security associations. These topics, among others for BGP, are important to know. However, they are not covered on this exam.

- BGP is an advanced link-state protocol
- BGP is used by every large ISP to connect with the Internet
- Two primary types of BGP relationships. It is important to remember that the default behavior changes with each.
  - Internal (with same AS)
  - External (with a different AS)
- BGP follows a set path selection criteria when choosing between multiple routes to the same destination
  - Highest ***Local Preference***
  - Shortest ***AS-Path***
  - Lowest ***Origin Code***
  - Lowest ***Multi-Exit Discriminator (MED)***
  - Prefer ***eBGP paths over iBGP paths***
  - ***Lowest IGP cost***, or metric (Closest next-hop address)
  - Prefer paths from the ***higher routing table number*** (inet.3 is preferred to inet.0)
  - Lowest ***router ID*** (or loopback IP address, if no router ID is specified) of the BGP router advertising the route
  - Prefer paths with the ***largest number of next hops***.
  - Prefer the ***lowest peer IP address***
- The problem with having to create a ‘full-mesh’ in order for BGP to function properly can be averted through the use of route reflectors or confederations
- Neighbors in “active” state are not UP!

Additional information (RFCs)

For specific information, please consult the following at http://www.ietf.org:

- RFC 827 and 904 EGP (1982/1984)
- RFC 1771, BGP version 4
- RFC 1772, Application of BGP in the Internet
- RFC 1773, Experience with BGPv4
- RFC 1774, BGP-4 Protocol Analysis
- RFC 1965, BGP Confederations
- RFC 1966, BGP Route Reflection
- RFC 1997, BGP Communities Attribute
- RFC 2439, BGP Route Flap Damping
- RFC 2796, BGP Route Reflection
- RFC 3065, AS Confederations for BGP
- RFC 2702, Requirements for Traffic Engineering over MPLS

Chapter Eight

MPLS

Chapter 8: MPLS

Overview

This chapter will cover *Multi-Protocol Label Switching* (MPLS). MPLS is a networking technology that integrates the control and robustness of IP routing with the simplicity and high-performance of layer-2 switching. MPLS enables service providers to create multiple, individual networks for diverse applications over a single IP infrastructure. Some of the important points you should be able to explain after reading this chapter:

- ✓ Define the purpose and value of deploying MPLS
- ✓ Clear understanding of the basics of *label distribution protocol* (LDP) and *resource reservation protocol* (RSVP)
- ✓ Name and define all label operations
- ✓ Difference in router types/functions in an MPLS network
- ✓ Traffic engineering within MPLS

Introduction

Different types of applications have varying network requirements for their data. File transfer and web pages are not sensitive to latency and are capable of compensating for packet loss during network events. Real-time voice and video traffic are extremely susceptible to performance degradation. The threshold for packet loss in such multimedia applications is miniscule, and even packets arriving out of order will result in incorrect presentation. In the past, service providers were often required to run separate networks for this sensitive data. *Asynchronous Transfer Mode* (ATM) provides stability and reliable *quality of service* (QoS) at the cost of scalability and additional network expense.

IP is inherently connectionless and in its original design did not provide controls for rigid QoS requirements. In order to attain multiple services over an IP network, that network must provide QoS levels equal to those offered by existing ATM and SONET technologies.

The original desire for MPLS was the need to increase the speed of layer-3 routing lookups to improve network performance. ATM switches performing this action via hardware are able to forward traffic faster than routers doing a longest IP match using software. This same functionality was desired in routers and could be performed using MPLS. In addition, MPLS has evolved into a protocol that provides for traffic engineering, *virtual private networks* (VPNs), and enhanced QoS.

MPLS is a method of encapsulating packets within a 32 bit header that is then used to make forwarding decisions.

MPLS Label Switched Paths (LSPs)

The distribution and application of MPLS labels creates *label switched paths* (LSPs). An LSP can be visualized as a tunnel which allows for the switching of packets across a network via the use of label. This eliminates the need for intermediate routers to perform IP route lookup. The entrance of the tunnel, or LSP, is referred to as the *ingress.* The exit of the LSP is referred to as the *egress*.

A router that places packets into an LSP by appending the packet with the MPLS header is known as the *ingress router*. At the LSP's exit, the router removing the label is known as the *egress router*. Routers that reside in the middle of an LSP between the ingress and egress are referred to as *transit routers*. In general, ingress and egress routers are referred to as *label edge routers* (LERs) because they reside at the edge of the MPLS network. Transit routers are often referred to as *label switch routers* (LSRs) as they perform switching operations to replace the incoming label with a new outgoing label within the core of the MPLS network. In other words, an LER will still do some type of action based upon the IP information of a packet. In contrast, a LSR acts upon only the MPLS information and is unaware of what IP information may be inside.

Using LSPs as next-hops for BGP is one example of how they can be used to improve network performance. Without their use, two IP lookups are required at each router hop to forward BGP traffic. The first lookup, which is based upon the packet's destination address, results in a BGP route. That result has a next-hop attribute that is the router ID of the BGP node that has the best route to the destination. The second lookup is done based upon the BGP next-hop and the result is an IGP route. After the second lookup, the packet is forwarded to the IGP next hop and the entire process is repeated.

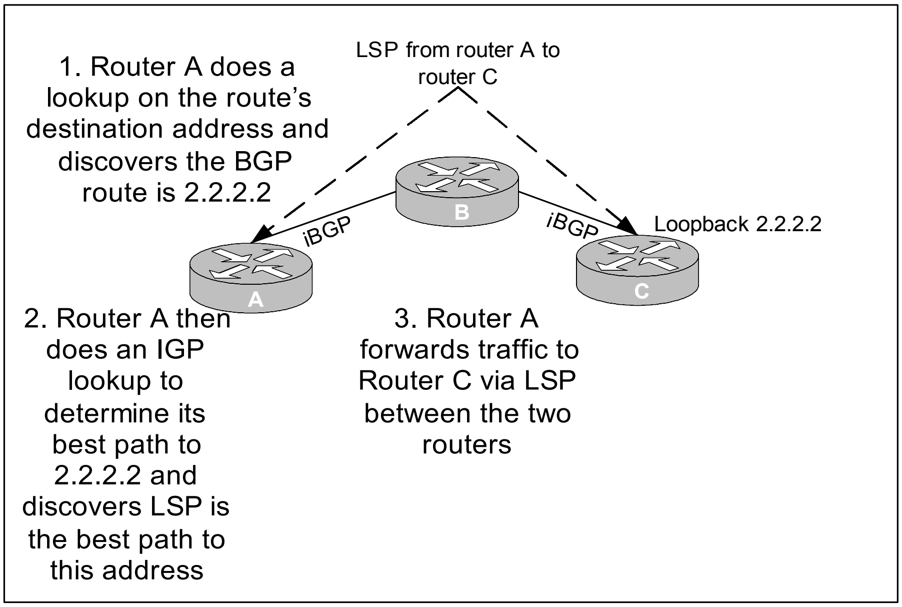

1. Router A does a lookup on the route's destination address and discovers the BGP route is 2.2.2.2
LSP from router A to router C
B
iBGP
iBGP
Loopback 2.2.2.2
A
C
2. Router A then does an IGP lookup to determine its best path to 2.2.2.2 and discovers LSP is the best path to this address
3. Router A forwards traffic to Router C via LSP between the two routers

Figure 8.1 BGP next-hop forwarding using an LSP

MPLS Header and Field Description

The MPLS header is 32 bits and consists of 4 distinct fields: *label*, *Class of Service* (CoS), *Stack,* and *time-to-live* (TTL). The label field of the MPLS header is 20 bits, the CoS field is three bits, the stack field is one bit, and the TTL field is eight bits.

20 bit Label	3 bit CoS	1 bit Stack	8 bit TTL
Field make-up of 32 bit MPLS header			

The label field is used when a packet is being MPLS switched. The lookup decision is based upon the label. The CoS field is used to classify the level of service the packet should receive. The stack bit is used to determine whether or not the top label is the only label appended to the packet. The TTL field defines the maximum number of hops the packet will be allowed to traverse.

The label space is made up of 20-bit unsigned integers between ranging from 0 to 1048575. 0 through 15 are reserved and have special functions. 16 to 1023 and 10000 to 99999 are unused and unassigned by JUNOS. These labels are best used for static LSPs because the dynamically assigned labels are not drawn from within this range. 1024 through 9999 are reserved for future applications. 100000 through 1048575 are dynamically negotiated, assigned, released, and reused by JUNOS.

Special labels are used for certain applications. One such instance is label distribution between the penultimate router (the last MPLS transit router) and the egress router. If it is desired that the penultimate router pop the label before forwarding the packet to the egress router the egress router would send a label of 3 to the penultimate router. Label 3 has a special meaning of "pop the label before forwarding". If it is desired that the penultimate router pop the label before forwarding the packet to the egress, the penultimate router would have label 3 assigned to the egress next hop. The other special labels are listed below with a brief explanation for each:

- 0, IPv4 Explicit Null Label—The label must be popped upon receipt and forwarding should continue based on the IPv4 packet address. (Note that this value is legal only when it is the sole label entry, no label stacking.)
- 1, Router Alert Label—When this is the top label, it should be popped and the packet delivered to the local software module for processing.
- 2, IPv6 Explicit Null Label—Same as label 0, except for IPv6
- 3, Implicit Null Label—Indicates penultimate hop router is next hop in the LSP
- 4 through 15—Reserved, but not yet assigned.

MPLS defines the process for switching packets that have labels. It is not a routing protocol and contains no ability to manufacture and propagate the labels themselves. Instead, it depends upon label distribution protocols to perform these functions. Similarly, at a high overview level, the basic IP routing process is concerned with only with doing a longest-match lookup and selecting the best next-hop. IP depends upon routing protocols such as OSPF and BGP to do the work of finding, advertising, and maintaining an accurate routing table.

Label Space: Allocation and Assignment

Label space is a range of label values that a router can assign or associate with a *forward equivalency class* (FEC). A FEC defines which packets will be appended with a particular label. One example of a FEC is a network address. All packets with a destination that belongs to the network address will be encapsulated with the same label before they are forwarded to the next router.

There are two ways a router can allocate its label space: *per interface* and *globally*. If a router is using the *per interface* method, a separate copy of the same label space is independently used for each interface. In other words, the same label can be distributed out each interface because the value of the label combined with the interface it was distributed out of gives it uniqueness. In contrast, with the global method all interfaces use the same label space. Therefore a unique label can only be distributed once. JUNOS employs the global method for label allocation.

Label Operations

A packet can be encapsulated with a number of labels organized as a last-in, first-out stack. This is referred to as the label stack. At a particular router, the decision as to how to forward a labeled packet is based exclusively on the label at the top of the stack.

There are three operations that can be performed to a label: *push, pop* or *swap*:

Push - Add a new label to the top of the packet. For IPv4 packets, the new label is the first label. If the push operation is performed on an existing MPLS packet, the result will be a packet with 2 or more labels. This is called label stacking. Any label other than the last label must have its *stack bit* set to 0 (the last label has its value set to 1). Note that in JUNOS software Release 4.2 and later, the new top label in a label stack always initializes its TTL to 255, regardless of the TTL value of lower labels.

Pop - Remove the top label from the packet. Once the label is removed, the TTL is copied from the label into the IP packet header, and the underlying IP data is forwarded as a native IP packet. In the case of multiple labels in a packet (label stacking), removal of the top label yields another MPLS packet. Note that in JUNOS software Release 4.2 and later, the popped TTL value from the previous top label is not written back to the new top label.

Swap - Replace the label at the top of the label stack with a new one. The stack bit is copied from the previous label, and the TTL value is copied and decremented (unless the *no-decrement-ttl* or *no-propagate-ttl* statements are configured). A transit router supports a label stack of any depth.

LSP Creation

Label distribution is the mechanism that creates LSPs. There are three ways labels can be distributed: *label distribution protocol* (LDP), *resource reservation protocol* (RSVP), and *static*. The method of label distribution dictates what type of LSP is formed. Each of the different types of LSPs has unique characteristics that dictate its capabilities, advantages, and disadvantages. Deciding which type of LSPs to deploy depends on the needs of the network. It is possible to deploy all three different types of LSPs concurrently. Static LSPs are rarely deployed on a large scale because of the administrative overhead.

Label Distribution Protocol (LDP)

LDP is used to dynamically create and exchange labels between MPLS/LDP routers. An LDP enabled router will assign labels to FECs, creating a label binding. Each router stores label bindings in a local database. Routers then exchange label databases via an LDP session. The label bindings are then used to encapsulate packets before forwarding them to the downstream LDP neighbor.

Two adjacent routers running LDP on a shared network segment will become LDP neighbors by exchanging discovery messages. Once two adjacent routers have become LDP neighbors they must establish an LDP session by exchanging session messages. The neighbor relationship exists at the interface level while the session is formed at the router ID level. If two routers are connected via two separate network segments they will become LDP neighbors on each segment while a single LDP session is formed between router IDs. Note the exception to this rule if the physical segments are independent (such as two ATM circuits each with its own VPI/VCI). In this exception, an LDP session will form over both of the physical links, independently.

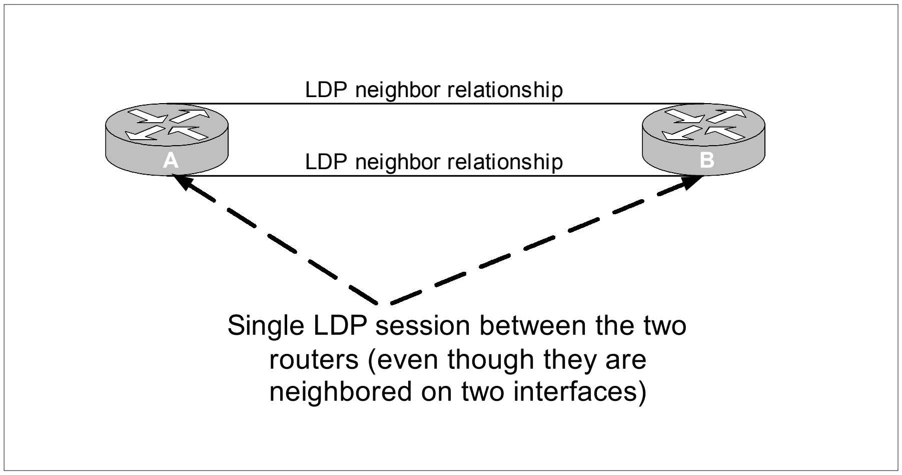

LDP neighbor relationship
LDP neighbor relationship
A
B
Single LDP session between the two routers (even though they are neighbored on two interfaces)

Figure 8.2 LDP neighbor adjacency and sessions

Two routers that form an LDP session whereby they exchange label bindings is the act of label distribution. This exchange or distribution of labels gives each router knowledge of what label to use when forwarding packets to the downstream router.

Labels are assigned by downstream routers relative to the flow of packets. A router receiving labeled packets (the next-hop router) is responsible for assigning incoming labels. A received packet containing a label that is unrecognized (unassigned) is dropped. For unrecognized labels, the router does not attempt to unwrap the label to analyze the network layer header, nor does it generate an ICMP destination unreachable message.

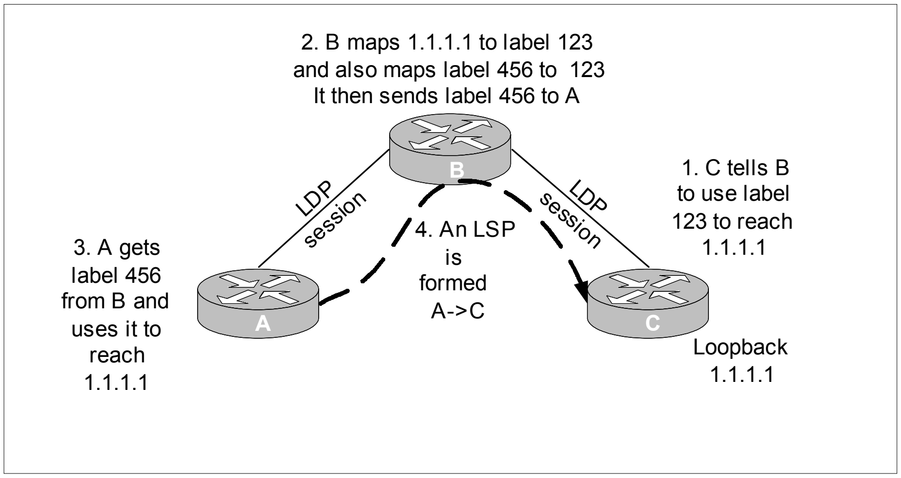

2. B maps 1.1.1.1 to label 123
and also maps label 456 to 123
It then sends label 456 to A
B
LDP session
LDP session
1. C tells B to use label 123 to reach 1.1.1.1
3. A gets label 456 from B and uses it to reach 1.1.1.1
4. An LSP is formed A->C
A
C
Loopback 1.1.1.1

Figure 8.3 Label distribution

For traffic to flow from Router A to Router C, each router needs to know which label the downstream router is expecting. The upstream routers know what labels to use because they are given this information by downstream routers. Keep in mind that when referring to MPLS and traffic flow, ‘downstream’ refers to the destination (router C in our example). ‘Upstream’ indicates the source (router A in our example). This is a simple concept when only dealing with three routers, but it is important to keep in mind when dealing with more complicated configurations.

Using the drawing in figure 8.3, let's assume that Router A receives a packet who's destination address is the loopback of Router C. Router A does a lookup and finds the label binding that it received from Router B (label 456), appends the appropriate label to the packet and forwards the packet to Router B. Because Router B told Router A what label to use, Router B knows that when it receives the packet it should swap the label with the label it received from Router C (label 123) and forward the packet to Router C.

Resource Reservation Protocol (RSVP)

RSVP is another type of LSP signaling protocol. It is used to distribute labels, reserve bandwidth, prevent loops, and inform the routers in the LSP of the type of traffic to expect. Two adjacent routers sharing a network segment running RSVP will become neighbors and exchange *path* (PATH) and *reserve* (RESV) messages when an LSP is signaled.

When an RSVP LSP is configured, PATH messages are sent toward the egress router informing each node along the way to prepare for a label binding, bandwidth reservation, and the type of traffic that will be sent along the LSP. When the egress router receives the PATH messages, it responds by sending a RESV message towards the ingress router using the same path. The RESV messages contain the label binding and amount of required bandwidth (as well as other information not pertinent to the JNCIA exam). Once RSVP has setup and signaled the path, the LSP is ready to be used for forwarding.

RSVP LSPs have the ability to signal a primary and secondary path. If network topology allows, the secondary path will not use any of the same hops as the primary. The secondary path is used for fail-over if the primary path should fail. Because the secondary path is pre-signaled and ready for traffic, it greatly decreases convergence times after a network event.

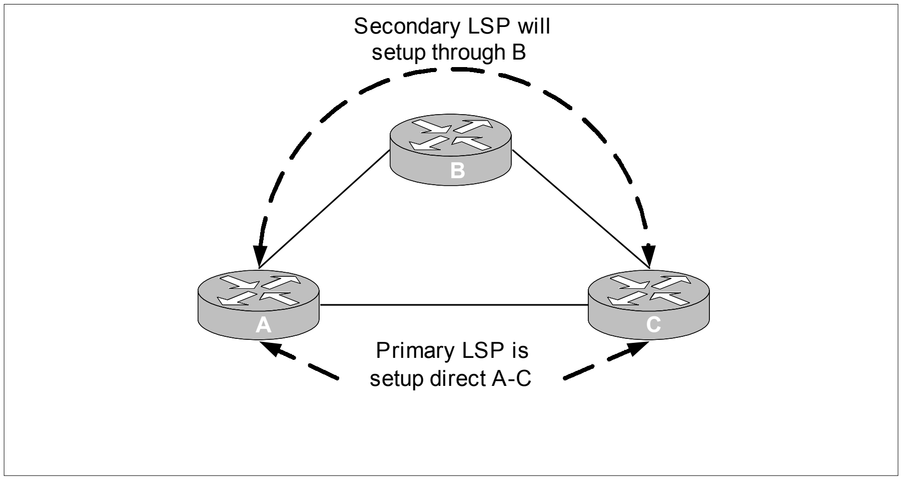

Secondary LSP will setup through B
B
A
C
Primary LSP is setup direct A-C

Figure 8.4 RSVP primary and secondary paths

RSVP LSPs can also be constrained. This is a form of traffic engineering. Any number of the hops within an LSP can be manually configured to constrain the path of the LSP. The remaining (non-constrained) hops are free to be signaled dynamically. There are two ways to configure the hops in a constrained path, *loose* or *strict*. When a path is configured with a strict hop, it must be directly connected to the previous hop. A loose constraint does not have to be directly connected to the previous hop but it must be traversed at some point within the LSP.

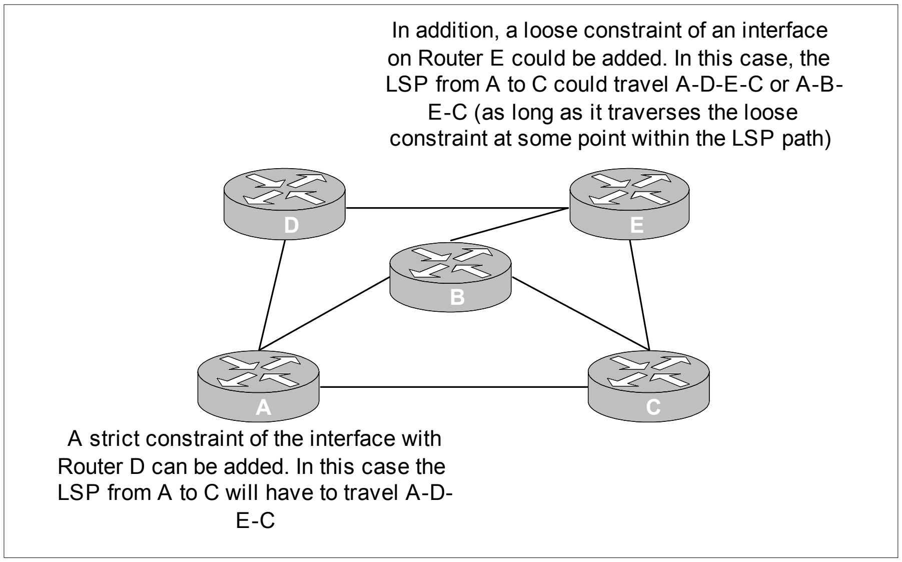

In addition, a loose constraint of an interface on Router E could be added. In this case, the LSP from A to C could travel A-D-E-C or A-B-E-C (as long as it traverses the loose constraint at some point within the LSP path)
D
E
B
A
C
A strict constraint of the interface with Router D can be added. In this case the LSP from A to C will have to travel A-D-E-C

Figure 8.5 RSVP LSPs with constrained hops

Traffic Engineering Database (TED)

The *traffic engineering database* (TED) contains information about the topology of the network. The TED is created by using the traffic engineering extensions to OSPF and/or IS-IS. In the case of OSPF, Type-10 LSAs are exchanged. IS-IS exchanges information within the link-state PDUs known as Type Length Values (TLV) that specify link attributes. The information about each node that is stored in the TED consists of:

- Node type
- Number of links
- Protocol that reported the information
- Static bandwidth
- ‘Reserveable’ bandwidth and
- Available bandwidth by priority level

Once each router's TED has been fully updated, this information can be used to calculate the best path for each LSP. An ingress router examines the requirements of an LSP and compares those against the information in the TED to calculate the best path. The result of the best path calculation is referred to as the e*xplicit route object* (ERO) and is passed on to RSVP to be used for signaling the LSP.

Constrained Shortest Path First (CSPF)

Constrained shortest path first (CSPF) is an algorithm similar to that used by OSPF and IS-IS to compute the shortest path between nodes. It takes LSP attribute information and compares it against the TED and compiles the best path for the LSP. The result of the CSPF computation is referred to as the ERO. The ERO is simply the interface addresses or hops the LSP will traverse to reach the egress router. CSPF passes the ERO to RSVP.

MPLS Configuration in JUNOS

Configuring MPLS can be quite complex and involved. Fortunately, the JNCIA does not focus on the configuration of MPLS as much as it does the operation of it. More emphasis is given to the editing in the higher level certifications. However, there are some facets with which you should be familiar.

Simply put, there are three distinct sections that must be handled to successfully configure MPLS. First, MPLS must be added at the global protocols level. Secondly, MPLS must be enabled on the interfaces it will run on. Third, the label protocols LDP and RSVP must be configured.

MPLS is enabled globally under protocols. All LSPs are configured under the `[edit protocols mpls]` layer. Traceoption files are also configurable within this level.

MPLS inherently does not route packets in the traditional sense, it forwards data depending on its label. In the same way, it is more a routed protocol than a routing protocol. Because of this, interfaces normally operating in the inet (internet) family must also be placed in the mpls family to be bound correctly to the protocols. This is done at the interface level of configuration under the `[family mpls]` level.

LDP must be enabled at the global level before it can be configured on each interface where it is desired. The top level of configuration is at `[edit protocols ldp]`.

Monitoring MPLS in JUNOS

Despite the relative complexity of configuring MPLS, troubleshooting and monitoring is no more difficult than dealing with another protocol. Viewing neighbors and interfaces for RSVP and LDP is straightforward and should be recognizable from working with the other routing protocols.

```
Show rsvp neighbor neighbor-IP <detail>
```

If the neighbor address is left out, an abbreviated list of neighbors will be given. The ‘detail’ option will return specific information on the queried neighbor(s).

```
jncia@my.router> show rsvp neighbor
RSVP neighbor: 6 learned
Address     Idle Up/Dn LastChange  HelloInt HelloTx/Rx MsgRcvd
10.159.1.34  0    1/0  2w3d 9:51:44     3      20/21        20
10.247.9.7   0    8/7  1w4d 8:53:38     3      35/34        34
10.247.9.106 0  28/27  3w5d 2:17:43     3      31/30        31
10.159.0.193 0    1/0  2w3d 5:38:28     3      79/80        80
10.159.0.161 0    1/0  4w3d 7:24:04     3      94/95        94
10.159.0.234 0  63/62  1w3d 2:55:01     3      27/29        29
```

```
Show rsvp interface <brief|detail>
```

Likewise, this command will output the status of the active RSVP interfaces, including bandwidth statistics for amount available, amount reserved, and highwater mark.

```
jncia@my.router> show rsvp interface
RSVP interface: 2 active
                Active Subscr- Static   Avail     Rsvd   High
Interface  State resv  iption    BW      BW        BW    mark
so-0/0/0.0  Up    18   100%   622.08Mb 605.3Mb   16.7Mb  1.4Mb
so-1/0/0.0  Up    12   100%   622.08Mb 605.3Mb   16.8Mb  1.3Mb
```

```
Show ldp neighbor <brief|detail|extensive>
```

Outputs the current status of LDP neighbors and the associated interface.

```
jncia@my.router> show ldp neighbor
Address       Interface  Label space ID  Hold time
10.1.8.1      lo0.0       10.1.8.1:0          10
10.1.11.2     lo0.0       10.1.11.2:0         13
10.1.11.25    lo0.0       10.1.11.25:0        10
```

```
Show ldp session <brief|detail|extensive>
```

Displays the status of LDP sessions to neighboring routers.

```
jncia@my.router> show ldp session
  Address    State        Connection    Hold time
10.1.2.2   Operational      Open            24
10.1.2.4   Operational      Open            26
```

When dealing with the state of label switch paths that have been set up through distributed labels:

```
Show mpls lsp [down|up|name] <detail>
```

```
jncia@my.router> show mpls lsp
Ingress LSP: 3 sessions
To           From        State Rt    ActivePath   P    LSPname
10.20.40.1   10.20.30.1  Up    9    to-router1   *    lsp2-R1
10.20.40.2   10.20.30.1  Up    8    to-router2   *    lsp2-R2
10.20.40.3   10.20.30.1  Up    6    to-router3   *    lsp2-R3
Total 3 displayed, Up 3, Down 0

Egress LSP: 3 sessions
To          From        State Rt Style Labelin Labelout LSPname
10.20.30.1  10.20.40.1  Up   0  1 FF      3        -  from-R1
10.20.30.1  10.20.40.2  Up   0  1 FF      3        -  from-R2
10.20.30.1  10.20.40.3  Up   0  1 FF      3        -  from-R3
Total 3 displayed, Up 3, Down 0
Transit LSP: 2 sessions
To          From        State Rt Style Labelin Labelout LSPname
10.20.40.1  10.20.40.3  Up   0  1 FF   182706  407173  R1-toR3
10.20.40.3  10.20.40.1  Up   0  1 FF   182701  407167  R3-toR1
Total 2 displayed, Up 2, Down 0
```

Note in the above ‘ingress’ output that all LSPs are ‘from’ the router running the command. The ‘P’ column denotes if the path is primary. Under all sections, the LSP name is listed in the final column. Likewise, under egress all LSPs are 'to' this router. Note the label in for egress LSPs is 3 and the outgoing label is undefined. This is because the packet will be IP forwarded after the final label is removed.

The 'detail' output of a specific LSP will show a wealth of information, including:

- The LSP metric
- The computed ERO with next-hops
- The computed CSPF metric cost
- Primary and Secondary paths

Below is the output from the RSVP log file with the 'state' traceoption flag set illustrating the steps undertaken while an adjacency forms.

```
jncia@my.router1> show log rsvp
RSVP new Session 10.2.2.2(port 18) Proto 0
RSVP new path state, session 10.2.2.2(port 18) Proto 0
RSVP new Neighbor 172.16.1.1
RSVP new resv state, session 10.2.2.2(port 18) Proto 0
RSVP neighbor 172.16.1.1 up on interface so-1/0/0.0
```

Key Points

MPLS is a solution for implementing stable and reliable QoS onto a packet switched network in a scalable fashion. It allows the building of converged networks where many different applications with different service level requirements utilize the same underlying network infrastructure.

- MPLS utilizes labels instead of IP addresses to forward data along predetermined label switch paths (LSPs).
- Multiple labels can be stacked onto a packet. Only the top label is acted on when the packet is analyzed.
- There are three basic functions carried out on an MPLS label:
    - Push
    - Swap
    - Pop
- Data traffic entering an LER undergoes and IP lookup which assigns a label to the packet. Thereafter until it exits the MPLS network the packet is label switched; LSRs pay no attention to the IP header of the packet.
- LDP and RSVP are the two protocols used by Juniper routers to produce and distribute labels and set-up LSPs
- Information about the network links is stored in the TED and used to determine the best path for LSPs
- The TED is compiled using information from the IGP:
    - OSPF Type-10 LSAs
    - IS-IS TLV extensions
- CSPF computes the best path given the information in the TED, the result is called the ERO.

In a non-MPLS network a packet enters a router, the router examines the packets header to learn the destination address. Then a lookup of the routing table is done based on the destination address and a BGP route is selected. The router then examines the BGP route and based on the BGP next hop does a second lookup. The result of the second lookup is used to forward the packet to the next router hop where both exhaustive longest match routing table lookups are done once again. This process is repeated until the packet arrives at the destination. It is the two lookup process that slows the networks throughput and overall performance.

In an MPLS network, LSPs can be configured between BGP speaking routers. As a packet enters the ingress router the initial IP lookup is done using the routing table. A BGP route is selected and the recursive lookup on the BGP next hop is also done. When the recursive lookup is done an LSP is found as the IGP next hop. This is the point where the ingress router encapsulates the packet with an mpls header and forwards the encapsulated packet to the next router. When the next router receives the packet the only lookup that needs to be done is based on the fixed length 20bit label contained in the MPLS header. The MPLS header is examined to learn the value of the label, a lookup is done in the label information base, a new label is appended to the packet and the packet is forwarded to the next router. Because each transit router only needs to perform a single lookup based on the label the end result is a network that can forward packets faster because it isn't necessary to perform two lookups at every router hop.

Additional Information

- RFC 3036, LDP Specification
- RFC 2209, Resource Reservation Protocol v1
- RFC 3209, RSVP-TE, Extensions for LSP tunnels
- RFC 3031, Multiprotocol Label Switching Architecture
- RFC 2702, Requirements for Traffic Engineering over MPLS
- RFC 3469, Framework for MPLS based Recovery
- RFC 3032, MPLS Label Stack Encoding
- RFC 2547, BGP/MPLS VPNs

Chapter Nine

Multicast

Chapter 9: Multicast

Overview

This chapter will cover the concepts and protocols used to implement multicast on Juniper routers. There are a number of unique ideas that will be covered. By the end of this chapter you should be able to:

- ✓ Site the benefits of implementing multicast.
- ✓ Describe multicast traffic flow.
- ✓ Identify multicast specific protocols DVMRP, PIM, MSDP, and IGMP.
- ✓ Understand the use of Reverse Path Forwarding.
- ✓ Recognize Class D address space.
- ✓ Define (S, G) and (*, G) notation.
- ✓ Recognize the use of multicast extensions to BGP.
- ✓ Identify the differences in PIM sparse-mode versus PIM dense-mode.
- ✓ Identify which routing table stores multicast routes.

Introduction

There are three fundamental ways to route IP packets: Unicast, Broadcast, and Multicast. Unicast traffic is the sending of packets from one unique source to one unique destination. The majority of the previous sections have dealt with the idea and concepts supporting unicast. Broadcasting is the concept of one unique source sending traffic to all destinations on a network. Broadcasting does not scale as network size increases; it rapidly uses up all available bandwidth resources.

The primary benefit of multicast is the conservation of bandwidth by allocating a single flow from a source to reach multiple destinations. In contrast, unicast must source a single flow **per** destination. Diagrams 9.1 and 9.2 illustrate this concept:

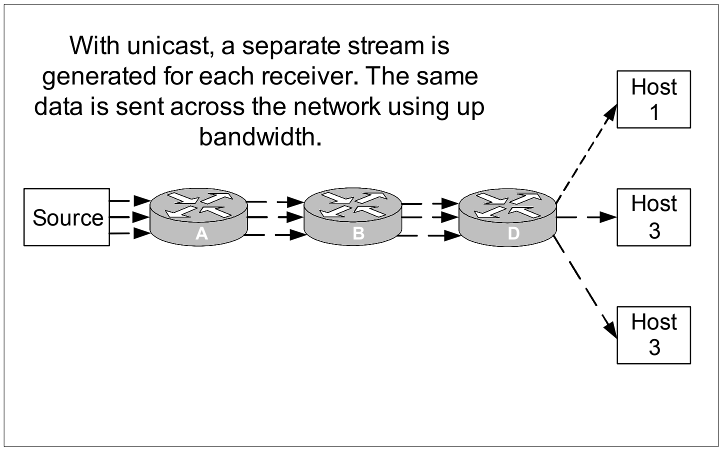

With unicast, a separate stream is generated for each receiver. The same data is sent across the network using up bandwidth.
Source
A
B
D
Host 1
Host 3
Host 3

Figure 9.1 Unicast Traffic Flow

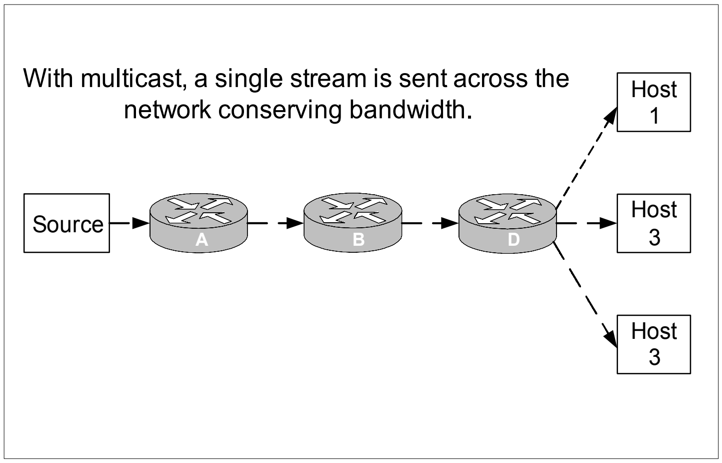

With multicast, a single stream is sent across the network conserving bandwidth.
Source
A
B
D
Host 1
Host 3
Host 3

Figure 9.2 Multicast Traffic Flow

In Figure 9.1, the data source must allocate an individual stream for each of the receiving hosts. Even though the path, and ostensibly the data payload, is the same, three flows must be managed. This rapidly depletes network resources as the number of hosts increases. In contrast, the source in 9.2 only needs to deliver a single stream to the last common device. In this case, Router D receives the single stream, replicates the packets and serves all three end-hosts.

The key to multicast resides within a number of specialized protocols and in the addressing. IP unicast traffic is sent to the unique host address allocated to that destination node. Broadcast traffic uses the special broadcast IP address, a .255 or "all 1's" subnet address. Multicast addresses are of a special subset of IP space within Class D. Routers running multicast protocols depend on addresses in Class D space to accurately deliver multicast packets only to devices that need them. Juniper routers store multicast routes in the inet.2 routing table, by default.

Multicast addressing

A single multicast address identifies a multicast stream. This is in contrast to a unicast address which identifies a unique host. This is an important differentiation to keep in mind. Multicast group addresses are taken from the Class D address space. This is defined as the addresses with the high-order bits set to '`1110`', giving an address range from `224.0.0.0` through `239.255.255.255`, or more simply `224.0.0.0/4`. All of the addresses in this range are reserved for multicast use. There are ranges within these class D addresses that are reserved and not available for public use much as there are ranges of IPv4 unicast addresses that are for private use.

IGMP

Internet Group Management Protocol (IGMP) dynamically controls the membership of hosts and routers for multicast groups. Hosts use IGMP to communicate their membership status to neighboring routers. Routers use IGMP to learn which sub-networks connected to them have multicast group members.

IGMP has two widely supported versions. Version 2 is backward compatible with Version 1, but when operating in a mixed environment the whole group must operate at the lower level. JUNOS software supports IGMP versions one and two. When IGMP is configured, version two is enabled by default.

IGMP Version 1

IGMPv1 utilizes two types of messages. Routers use a membership *query* and hosts use the membership *report*. The IGMP enabled router will periodically send query messages to the hosts on locally attached subnets soliciting multicast group membership. Queries are sent on the multicast address 224.0.0.1 with a TTL of 1 to ensure only connected hosts receive the request. When a host first wishes to join a multicast stream, it sends an unsolicited report message to the LAN router. This prevents delays while waiting for the next router query before beginning to receive multicast traffic. The hosts on the subnet who are members of a multicast group continue to periodically send report messages to let the upstream multicast router know that they still want to be a group member. Only a single host under the router needs to send a report to keep the multicast group active. A report from one host will suppress report messages from the others.

The router continues to send queries as long as it receives report messages from the hosts indicating that the multicast stream is still required. Version one hosts silently leave the group, meaning that an end system that no longer desires the multicast stream will cease sending report messages. The router will continue to forward multicast packets to all host interfaces as long as there is an active group. Hosts that are not part of the multicast group simply discard the packets. When all hosts stop responding to router queries with report messages, the entire multicast group times out and the stream is stopped.

IGMP Version 2

IGMPv2 adds a finer grain of control over group membership through the use of new message types. IGMPv2 routers utilize group specific queries. Hosts use *join* and explicit *leave* messages as well as report messages. When a version two host requires a multicast stream, it sends a join message to the upstream router requesting group membership.

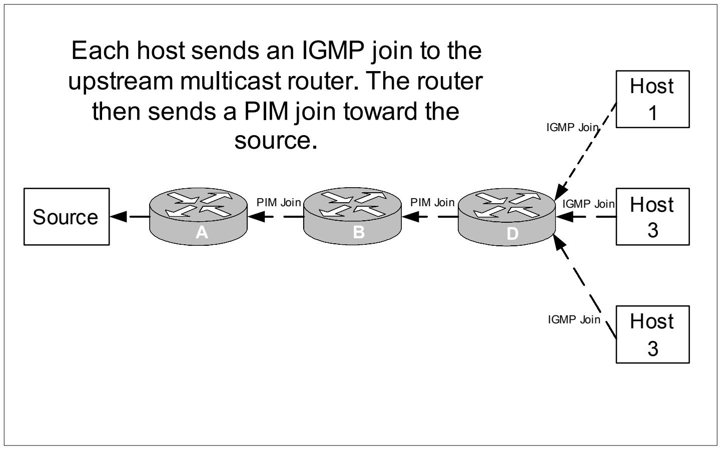

Each host sends an IGMP join to the upstream multicast router. The router then sends a PIM join toward the source.
Host 1
IGMP Join
Source
PIM Join
PIM Join
IGMP Join
Host 3
A
B
D
IGMP Join
Host 3

Figure 9.3 IGMP Join Behavior

Hosts continue to maintain the group by sending report messages to the router, notifying it that the stream is still needed. When a host chooses to no longer be a member of the multicast group, a leave message is sent to the upstream multicast router to let the router know that this host no longer wants to receive the multicast data stream.

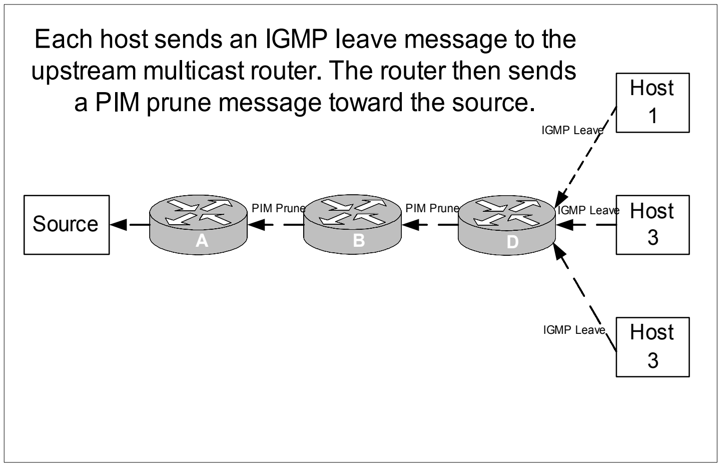

Each host sends an IGMP leave message to the upstream multicast router. The router then sends a PIM prune message toward the source.
Host 1
IGMP Leave
Source
PIM Prune
PIM Prune
IGMP Leave
Host 3
A
B
D
IGMP Leave
Host 3

Figure 9.4 IGMP Leave Message

When an IGMPv2 router receives a leave message, it queries the group to make certain there are still active members. If it does not receive a host report, the group times out and the stream is pruned. This reduces the overall latency involved with version one hosts silently dropping group membership.

DVMRP

Distance Vector Multicast Routing Protocol (DVMRP) dynamically generates shortest path forwarding trees to deliver multicast packets. It is designed to be the IGP of a multicast domain. By dynamically generating IP multicast delivery trees, DVMRP provides connectionless datagram delivery to host members of the group. Through these techniques, shortest path trees are created and used to send multicast data streams to group members.

DVMRP neighbors are discovered dynamically by sending *probe* messages periodically to the IP multicast group address that is reserved for all DVMRP routers (224.0.0.4), similar to how OSPF hellos are sent to the IP multicast group address that is reserved for all OSPF routers.

DVMRP has two methods for routing: *forwarding mode* and *unicast routing mode*. In forwarding mode, it performs unicast routing and IP multicast data forwarding. In unicast routing mode, DVMRP performs unicast routing, but to forward IP multicast data *Protocol Independent Multicast* (PIM) must be enabled on the outgoing interface. Because PIM has its own method of discovering best path routing, the default DVMRP mode is forwarding.

Reverse Path Forwarding (RPF)

All of the major IP multicast routing protocols make use of a *distribution tree* to forward data. There are two different kinds of distribution trees. A *shared* tree is created by mapping the best logical path from the client to the *rendezvous point* (RP). The concept of an RP is integral to PIM and will be covered in a following section. A *source* tree is created by mapping the best logical unicast path from the multicast source to a group receiver or member. A technique called *Reverse Path Forwarding*, or RPF, is used to prevent the resending of IP multicast data back up the distribution tree.

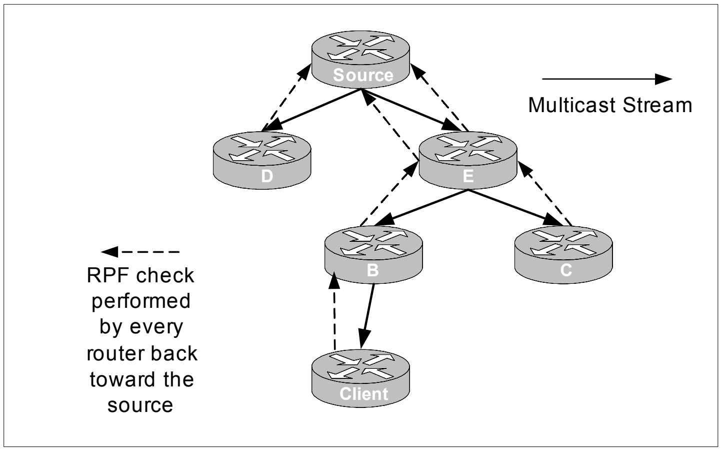


Figure 9.5 RPF Check

We will now discuss how reverse path forwarding functions. We will start with examining the RPF mechanism for data flowing down a source tree:

1) The router examines the source unicast IP address of the arriving multicast packet to determine whether the packet arrived via an interface that is on the path **back** to the source (the reverse path).
2) If the packet arrives on the interface back to the source, the RPF check is successful and the packet is forwarded out all multicast enabled interfaces except the one on which it was received.
3) If the RPF check fails, the packet is silently discarded.

All routers in Figure 9.5 pass an RPF check. This is because the multicast stream is coming into the same interface the router would use to send unicast packets to the source. However, if there were a connection between routers B and C, both of these routers would forward the stream out those connected interfaces. That flow, from B to C and likewise C to B, would fail the RPF check. The reason behind the failure is that both B and C look to router E when checking the unicast path back to the source, not to each other.

Next, we will examine the process as it is carried out for a shared tree (from the multicast source to the client):

1) The router examines the address of the arriving multicast packet and determines the RP associated with it.
2) If the packet arrives on the interface back to the RP, the RPF check is successful and the packet is forwarded.
3) If the RPF check fails, the packet is silently discarded.

Source and Group (S, G)

Because a multicast address refers to a stream rather than a host, routers must keep track of interfaces that forward multicast traffic and the interface where the multicast stream is received. This is accomplished by examining the IP address of the source (S) and the IP multicast group address (G). These two combine into what is called (S, G) notation. If the source is not known, but the group is available, the pair is referred to as (*, G). This is common when a client requests a multicast stream for the first time, as the source would not be known.

Looking at some simple examples to illustrate, multicast stream 224.53.20.16 originating from host 10.100.23.65 will have the (S, G) notation:

(10.100.23.65, 224.53.20.16)

A host joining group 224.53.20.16 for the first time, without knowledge of the source, has the notation:

(*, 224.53.20.16)

Sparse-mode vs. Dense-mode

IP multicast routing protocols generally follow one of two basic approaches depending on the expected distribution of multicast group members throughout the network.

The first approach is based on assumptions that the multicast group members are densely distributed throughout the network (i.e., many of the subnets contain at least one group member) and that bandwidth is plentiful. So-called "dense-mode" multicast routing protocols rely on a technique called *flooding* to propagate information to all network routers. Dense-mode routing protocols include Distance Vector Multicast Routing Protocol (DVMRP), Multicast Open Shortest Path First (MOSPF), and Protocol-Independent Multicast - Dense Mode (PIM-DM). MOSPF is not critical to the JNCIA exam and will not be covered in depth.

The second approach to multicast routing basically assumes the multicast group members are sparsely distributed throughout the network and bandwidth is not necessarily widely available. Sparse-mode does not imply that the group has a few members, just that they are widely dispersed. In this case, flooding would unnecessarily waste network bandwidth and could cause serious performance problems. Hence, sparse-mode multicast routing protocols must rely on more selective techniques to set up and maintain multicast trees. Sparse-mode routing protocols include Core Based Trees (CBT) and *Protocol-Independent Multicast-Sparse Mode* (PIM-SM). CBTs are not supported by JUNOS and will not be covered on the JNCIA exam.

Protocol Independent Multicast (PIM)

PIM gets its name from the fact that it is not reliant upon any specific IGP. That is, regardless of which unicast routing protocol(s) is (are) used to populate the unicast routing table, PIM uses this information to perform multicast forwarding.

Two implementations of PIM exist: PIM Sparse Mode (PIM-SM) and PIM Dense Mode (PIM-DM).

Concepts common to PIM-DM and PIM-SM include:

- PIM Neighbor Discovery.
- PIM Asserts.
- Protocol independent.
- Source Path Tree (SPT).
- No separate multicast routing protocol.

Some key characteristics of PIM-SM are:

- Explicit Join behavior.
- Utilization of Rendezvous Point (RP).
- Rendezvous-point Path Tree (RPT).

PIM Neighbor Discovery

Like DVMRP, PIM uses a neighbor discovery mechanism to establish neighbor adjacencies. To establish these adjacencies, a PIM multicast router multicasts a PIM Hello message to the address 224.0.0.13

(signifying All-PIM-Routers) on each of its multicast enabled interfaces. The interval for these advertisements is known as the *hello-period,* and is set to 30 seconds by default.

PIM Hello Messages

PIM Hello messages contain a hold time value, which tells the receiver when to expire the neighbor adjacency associated with the sender if no further PIM Hello messages are received. The value that is sent in the hold time field is typically three times the sender's PIM hello-period, or 90 seconds by default.

PIM Designated Router

In addition to establishing PIM neighbor adjacencies, PIM Hello messages are also used to elect the Designated Router, or DR, for a multi-access network. The router with the highest multicast enabled IP address becomes the DR for the network. If the DR is already established and another router joins the multi-access network with a higher IP address, an election will not be triggered.

PIM Sparse Mode

PIM-SM, like PIM-DM, uses the unicast routing table to perform the RPF check function instead of maintaining a separate multicast routing table.

Rendezvous Point (RP)

The RP serves as the information exchange point for other multicast routers. The RP is the only router with knowledge of all active sources in a domain. The remaining routers need only know a unicast path back to the RP.

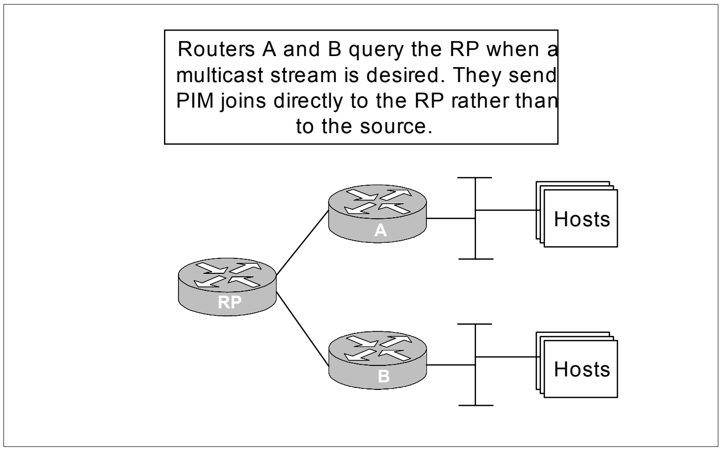

Routers A and B query the RP when a multicast stream is desired. They send PIM joins directly to the RP rather than to the source.
A
Hosts
RP
B
Hosts

Figure 9.6 Rendezvous Point

Rendezvous-point Path Tree (RPT)

The path between an RP and its receivers is known as the RPT. In PIM-SM, one requirement is that the multicast traffic must traverse the RP. To force traffic through the RP, an RPT must be established between the receiver's designated router and the RP.

After a designated router has received an IGMP join, it sends a PIM join toward the RP. This ensures that multicast data will traverse through the RP before being sent on to the designated router and its directly attached receivers.

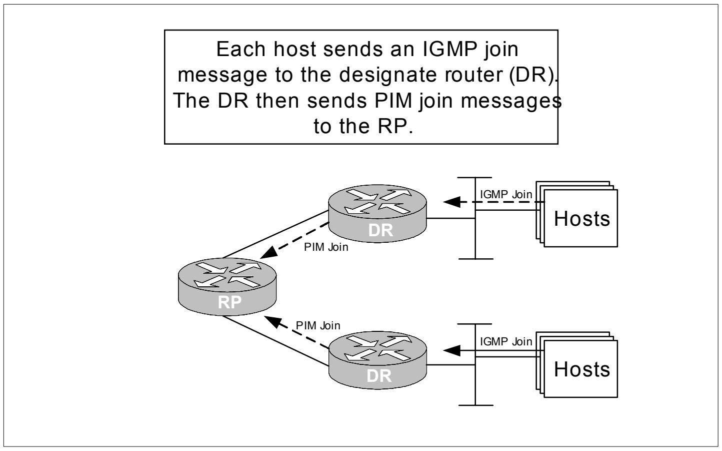

Each host sends an IGMP join message to the designate router (DR). The DR then sends PIM join messages to the RP.
IGMP Join
DR
Hosts
PIM Join
RP
PIM Join
IGMP Join
DR
Hosts

Figure 9.7 RP traffic flow behavior

Explicit Join Model

PIM-SM conforms to the sparse-mode model where multicast traffic is only sent to locations of the network that specifically request it. In PIM-SM, this is accomplished via *PIM Joins*, which are sent hop-by-hop toward the root node of the tree. (Note: In the case of the shared tree the root node is the RP. Conversely, in the case of the shortest path tree the root node is the first hop router directly connected to the multicast source.) As this Join travels up the tree, routers along the path set up multicast forwarding state so that the requested multicast traffic will be forwarded back down the tree to the receiver.

Likewise, when multicast traffic is no longer needed, a router sends a *PIM Prune* up the tree, toward the root node, to prune off the unnecessary traffic. As this PIM Prune travels hop-by-hop up the tree, each router updates its forwarding state appropriately. This update often results in the deletion of the forwarding state associated with a multicast group or source.

PIM-SM Shared Trees

PIM-SM operation centers around a single, unidirectional shared-tree whose root node is the RP. These shared trees are sometimes called RP trees because they are rooted at the RP. Last-hop routers (i.e. routers with a directly connected receiver for the multicast group) that need to receive the traffic from a specific multicast group, join this shared tree. When the last-hop router no longer requires the traffic of a specific multicast group, the router prunes itself from the shared tree.

Shared Tree Joins

Figure 9.8 shows the first step of a Shared Tree Join in a sample PIM-SM environment. In this step, a single host, Receiver 1, has just joined multicast Group G, via an IGMP report.

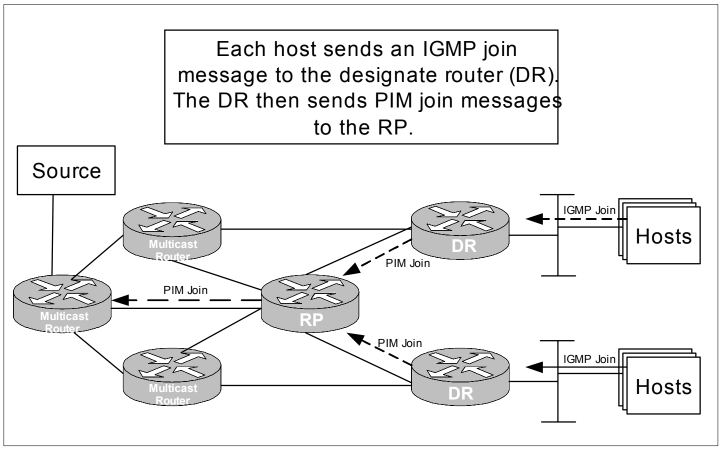

Each host sends an IGMP join message to the designate router (DR). The DR then sends PIM join messages to the RP.
Source
Multicast Router
Multicast Router
Multicast Router
RP
DR
DR
PIM Join
PIM Join
PIM Join
IGMP Join
IGMP Join
Hosts
Hosts

Figure 9.8 Shared Tree Join

Source Path Tree (SPT)

Because it is not always optimal to force multicast traffic through an RP, an SPT is built without one. In these cases, an SPT is used because it allows a path to be established directly between receivers and the source.

Instead of the designated router sending PIM joins to an RP, they are sent directly to the source. Because the PIM join message is directed toward the source, the concept of an RP becomes irrelevant. This ensures that multicast traffic will use the best path when being delivered to receivers without the restriction of being routed through an RP.

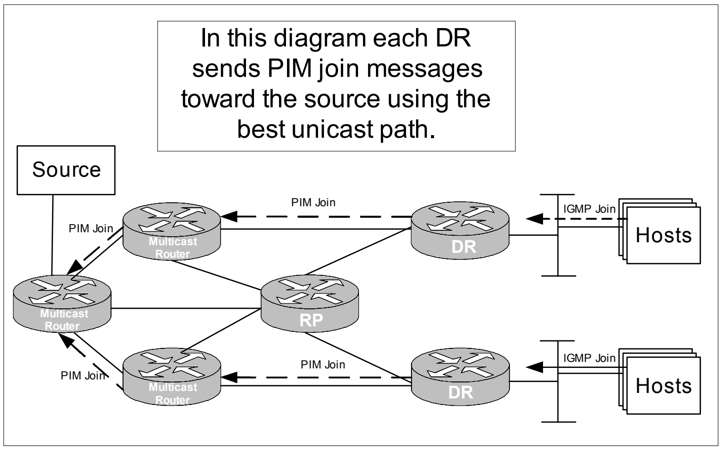


Figure 9.9 Source Path Tree

PIM Join/Prune Messages

PIM Join and PIM Prune messages use the same message format type. The messages contain the following fields:

- Multicast Source Address – IP address of the multicast source to Join or Prune. (If the Wildcard flag is set, this is the address of the RP.)
- Multicast group address - Class D multicast address to Join or Prune.
- WC bit (Wildcard flag) – Indicates that this entry is a shared tree (*, G) Join or Prune message
  RP bit (RP Tree flag) – Indicates that this Join or Prune information is applicable to, and should be forwarded up, the shared tree.

Each PIM Join or Prune contains both a Join list and a Prune list. Either one may be empty, depending on the information being conveyed up the distribution tree. By including multiple entries in the Join and/or Prune lists, a router may Join or Prune multiple sources and/or groups with a single message.

For example, examine the below PIM Join and Prune message:

Source address = 128.247.1.1

Group Address = 224.1.1.1

Flags = WC, RP

This Join message indicates that this item is a (*, G) Join for group 224.1.1.1 with an RP of 128.247.1.1. Note that since the RP flag is set, the source address is that of the RP and not the actual source of the multicast stream.

PIM-SM Designated Router

On a multi-access LAN running PIM-SM, the DR has additional responsibilities. The first is sending Joins to the RP to construct the shared tree. When more than one router is connected to a LAN segment, PIM-SM provides not only a method to elect the DR but also a way to detect the failure of the current DR. If the current DR were to fail the routers on that LAN would lose their adjacencies with the DR when they timed out. When this happens an election takes place and a new router becomes DR. Since each PIM-SM router is responsible for maintaining states from IGMP Membership Reports, the new DR will immediately send a Join to the RP for the appropriate multicast groups.

Multiprotocol Border Gateway Protocol (MBGP)

MBGP is an extension to BGP that allows it to carry multicast routing information used for RPF calculations between autonomous systems. MBGP is implemented as two additional BGP attributes: MP REACH NLRI and MP UNREACH NLRI. These attributes are used to exchange reachability information for different address families and are carried inside BGP update messages. Contained within the attributes are the *Address Family Identifier* (AFI) and the *Subsequent Address Family Identifier* (Sub-AFI) which identify the protocol for which the reachability information is applicable. There is far more information regarding MBGP, however it is not critical to the JNCIA exam.

Multicast Session Discovery Protocol (MSDP)

MSDP allows RPs from different multicast autonomous systems to exchange source and group (S,G) information. Two RPs from different multicast autonomous systems that become MSDP neighbors will exchange *Source Active* (SA) messages so that each can learn of active sources residing outside of the local multicast domain.

In order to prevent loops, MSDP relies on MBGP and the underlying IGP for all forwarding decisions. The decision making process can be broken down into two rules:

- An internal MSDP peer will accept SA messages from another internal peer only if that peer is the closest advertiser to the RP for the prefix covering the RP that originated the SA message.
- An external MSDP peer will accept an SA message from another peer if and only if the peer is in the next AS in the path towards the AS originating the prefix covering the RP in the SA message.

Key Points

Multicast is a bandwidth-conserving technology that reduces network traffic by simultaneously delivering a single stream of data to multiple receivers.

- Multicast traffic utilizes Class D address space, from 224.0.0.0 to 239.255.255.255.
- IGMP is used to dynamically control multicast group membership.
  - Routers send Query messages to hosts.
  - Hosts send Join and Leave messages to their upstream router.
- DVMRP dynamically generates shortest path trees for multicast streams.
- RPF checks ensure multicast data is not sent back to the stream source.
- Dense-mode protocols assume many subnets contain at least one multicast group member and that bandwidth is plentiful. They rely flooding to propagate information to all network routers.
- Sparse-mode protocols assume multicast group members are widely dispersed and bandwidth is not necessarily available. They rely on selective techniques to set up and maintain multicast trees.
- (S, G) notation binds a source address with a multicast group address.
- (*, G) notation is used when the source is unknown.
- PIM-Sparse Mode utilizes a Rendezvous Point to facilitate host members receiving multicast streams.

Additional Information

- RFC 1075 Distance Vector Multicast Routing Protocol
- RFC 1112 IGMP Version 1: Host Extensions for Multicast
- RFC 1458 Requirements for Multicast Protocols
- RFC 1584 Multicast Extensions to OSPF
- RFC 2201 Core Based Tree (CBT) Multicast Routing Architecture
- RFC 2236 IGMP Version 2
- RFC 2283 Multiprotocol Extensions for BGPv4
- RFC 2362 Protocol Independent Multicast – Sparse Mode
- RFC 2715 Interoperability Rules for Multicast Protocols

Chapter Ten

Policy

Chapter 10: Policy

Overview

This chapter will cover JUNOS policy and firewall. The topics covered will include the purpose, functionality and configuration of policies and route filters. By the end of this chapter you should be able to:

- ✓ Identify the components of JUNOS policy.
- ✓ Understand how policy controls the exchange of routing information.
- ✓ Comprehend how route manipulation through policy dictates traffic flow.
- ✓ Recognize the purpose and application of the JUNOS firewall.
- ✓ Describe the differences between import and export policy
- ✓ Understand route redistribution.

Introduction

Each routing protocol run within JUNOS maintains its own routing database. JUNOS takes information from each protocol's database and generates the best forwarding path to each destination. Juniper policy allows you to control the routing information exchanged between routing databases. You can think of configured policies as a way to filter route advertisements; this is functionally similar, but structurally quite different than Cisco IOS access-lists. Keep in mind that an entire book could be written on JUNOS policy itself. We have attempted to cover general theory in addition to highlighting key points that are needed for the JNCIA exam.

JUNOS firewall is a policy whose function is to limit or manipulate traffic allowed through specific interfaces. The most common purpose of the firewall is to provide security for the Juniper router and downstream nodes and networks.

Policy Definition

Policies are made up of four key elements: *terms*, *from*, *then*, and 'actions'. The use of these establishes the groundwork for functional policy. An individual policy can become complex through the combination, repetition, and application of these four simple attributes.

Every policy begins by being named and defined with a *policy statement*. When viewing a policy from the CLI, it is referenced by the policy statement. The policy statement only names the policy; it does not perform an active function. For this reason, it is usually given an expressive name to highlight the purpose or function of the entire policy, for example:

```
jncia@my.router1>show policy reject-private-address-space
Policy reject-private-address-space:
    Term 1918-space:
        from
               route-filter:
                     10.0.0.0/8 orlonger
        then reject
    Term accept:
        then metric 100000 localpref 100
community + region community
```

Terms are the subroutines of a policy. Specific instructions are carried out within a term to *accept*, *reject*, or modify the attributes of routes. Depending on complexity, a policy can be made up of one or more terms. It is important to realize that a policy will begin analyzing candidate routes from the first term, and continue its way sequentially down the list of terms until it is instructed to exit or continue on to the next policy. If no match and subsequent action is taken within the policy, the default action is to accept the route.

From statements are the conditional clauses of the policy. Those familiar with programming languages can think of a *from* clause equating to an ‘if’ statement. If the criteria after the *from* stanza are met, the candidate route continues down the same term. If the criteria of the match are not met, the candidate jumps to the next term (if present). In order for JUNOS to take action on a route, it must **exactly** match the set specifications. If there is more than one criteria in a *from* clause, they must all be matched. Routes can be matched based on any of the following attributes:

- Source Address
- Destination Address
- Protocol (BGP, OSPF, ISIS)
- Port
- Community
- Route filters
- Interface
- Local Preference
- Others

‘Then’ statements come after the ‘from’ and are immediately followed by actions. The purpose of a ‘then’ is to indicate what action is performed after a match is made.

Actions are the operators of a policy. They are the instructions that allow the policy to *accept*, *reject*, or modify route attributes. Actions include:

- Accept – Permits the route.
- Reject – Denies the route.
- Next-policy – Moves on to the next policy.
- Next-hop-self – Sets the BGP next hop to this router.
- Community – Sets BGP community information.
- Localpref – Sets BGP local preference value.
- Metric – Sets BGP MED value.
- Count – increment counters of the specified name to indicate a match has taken place.

Note that there are more actions that can be performed with relation to other routing protocols, but they are not covered in detail for the exam.

Prefix Lists and Route Filters

One of the most common protocol-independent methods of identifying traffic within a policy is through the use of prefix lists and route filters. Prefix lists merely identify a route and its mask to form a match condition. An example of a prefix list is displayed below.

```
[edit]
jncia@my.router1> show policy-options
prefix-list match-slash-24-private {
   10.0.0.0/24;
}
```

This prefix list will match only one route: 10.0.0.0/24.

Route filters are similar in function, but they allow for more complex matches within a single statement. In addition, a route filter can immediately perform an action on a route that is matched. It is important to remember that “longer” refers to a longer prefix mask length. This is to state that a mask of /10 is longer than that of a /9 as it masks more of the address. One of the following terms may be contained by which to match a route:

exact- Exactly match the prefix length.
(*10/8 exact* matches only 10.0.0.0/8, nothing else)

longer- Mask is greater than the prefix length
(*10/8 longer* matches 10.0.0.0/9 and up, but not 10.0.0.0/8)

orlonger- Mask is equal to or greater than the prefix length
(*10/8 orlonger* matches 10.0.0.0/8 and up)

prefix-length-range- Mask falls between two prefix lengths
(*10/8 prefix-length-range /20-/24* will match 10.x.x.x if the mask is /21, /22, or /23,)

upto- Mask falls between two prefix lengths, including the upper limit
(*10/8 upto /12* will match 10.x.x.x if the mask is /9, /10, /11, or /12)

through- Route must fall between the two specified prefixes
(*10.0.0.0/8 through 10.192.0.0/10* will match 10.x.x.x/8, 10.0.x.x/9, 10.128.x.x/9, and 10.192.0.0/10)

As noted previously, a route filter can also specify actions immediately after matching a route. This will be seen in the example below. In addition, if no immediate action is specified, the matching routes will be subject to that term's "then" clause.

```
[edit policy-options]
jncia@my.router1# show

policy-statement route-filter-example {
  term match-some {
      from {
           route-filter 10.0.0.0/8 exact;
           route-filter 192.168.0.0/16 upto /19
reject;
      }
           route-filter 172.16.0.0/10 through
172.32.0.0/11 {
                metric add 1000;
           }
      }
      then {
           accept;
      }
}
```

Policy Lists

Another aspect of policies worth mentioning is the ability to note several match conditions (either AS path, community, interface, neighbor, next-hop, or protocol) in one statement with the use of a list. Only one of the members of the list must be matched for successful attachment of the statement.

```
   [edit policy-options]
   jncia@my.router# show
   policy-statement match-2 {
       term from-bgp {
            from {
                protocol bgp;
                neighbor [ 10.0.0.1 10.0.0.128 ];
                route-filter 172.16.0.0/10 through
172.32.0.0/11 {
                    metric add 1000;
                }
            }
            then accept;
       }
       then reject;
   }
```

Policy Evaluation

- Default behavior is to start with the first term of a policy.
- If no match is made, no action is taken in the current term; the route is evaluated against the next term in the policy
- If a match is made, an action is taken to accept or reject the route; the route will exit policy evaluation unless *next-policy* is specified as an action for this term
- If the candidate has not matched any terms within the policy, the next applied policy is evaluated (in the same term-by-term fashion)
- Upon reaching the end of the last applied policy, if no action is defined, the default action is taken (default action varies by protocol)

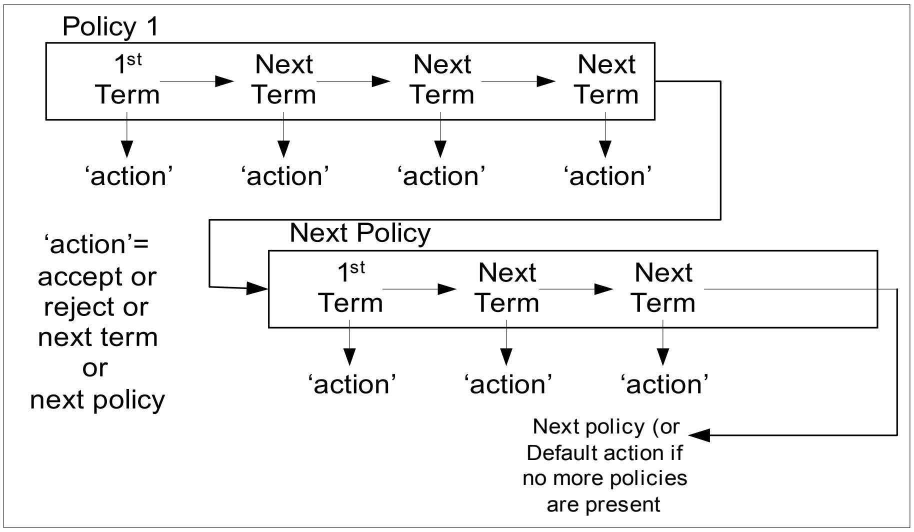


Figure 10.1 Policy evaluation flow

As noted, if a route makes it to the end of the last policy and is not accepted or rejected by that point, the default action will take place for that given protocol. The default actions for the most common routing protocols are listed in the table below. Note that they are listed in order of their level of importance to remember.

Protocol	Default Import Policy	Default Export Policy
BGP	Import all BGP routes	Export all active BGP routes
OSPF	Import all OSPF routes (Note: you cannot override this)	Export all active OSPF routes Export direct OSPF interface routes
RIP	Import all RIP routes	Nothing is exported
Pseudo-protocol (direct, static, aggregate, etc.)	Import all of these types of routes.	Nothing is exported A routing protocol is needed to export these.
IS-IS	Import all IS-IS routes (Note: you cannot override this)	Export all active IS-IS routes Export direct IS-IS interface routes
MPLS	Import all MPLS routes (into inet.3)	Export all active MPLS routes
LDP	Import all LDP routes (into inet.3)	Export all active LDP routes
DVMRP	Import all DVMRP routes (into inet.1)	Export all active DVMRP routes
PIM Sparse Mode	Import all PIM-SM routes (into inet.1)	Export all active PIM-SM routes
PIM Dense Mode	Import all PIM-DM routes (into inet.1)	Export all active PIM-DM routes

Figure 10.2 Default protocol policy actions

Let's examine the path a sample BGP route takes as it is evaluated against a policy designed to limit the number of advertisements from a BGP peer. This policy is applied as an import policy to the protocol BGP.
Route advertisement 10.64.65.66/32 from BGP neighbor 10.1.2.3.

The first step is to create a policy to ensure we only accept routes from the intended neighbor. The next step is to create a policy to only match the route we want to accept. These are illustrated below:

```
[edit policy-options]
jncia@my.router1# show
policy-statement from-10.1.2.3 {
    term match-neighbor {
        from neighbor 10.1.2.3;
        then accept;
    }
    term reject-rest {
        then reject;
    }
}
policy-statement match-route {
    term good-one {
        from {
            protocol bgp;
            next-hop 10.64.65.66/32;
        }
        then accept;
    }
    term reject-rest {
        then reject;
    }
}
```

The last step is to apply our ‘good-one’ filter to BGP so that we only accept this one route. This is shown below:

```
[edit protocols bgp]
jncia@my.router1# show
import match-route;
```

Application of Policy to Routing Protocols

When implementing routing policies, it is important to remember the "direction" of the policy application. Whether importing or exporting, it is from the perspective of the routing table. For instance, if routes are being sent to a BGP neighbor, they are being exported from the routing table. Conversely, if routes are received from a BGP peer, they are imported into the routing table. To illustrate:

Import: All unicast routing protocols place all of their routes into the main routing table (inet.0). If multiple routes exist to the same destination, the best one will be chosen via a routing algorithm (or multiple next-hops will be chosen if equal cost is determined).

Export: By default, routing protocols export only routes that were learned from **that** protocol (for example: BGP routes are exported to BGP). One supplement to this is the idea that OSPF and IS-IS export the direct routes (interfaces) on which they are configured.

Policy can be applied to routing protocols at the following levels and manner:

BGP: Global, group, and peer import/export.
(Note that the more specific application will override the more general. For example, policy applied to a peer will override group policy and group policy will override global application.)
IS-IS: Global export.
OSPF: Global export.
RIP: Neighbor import and group export.
DVMRP: Global import and export.

Testing Policy

JUNOS allows evaluation of policy syntax and basic logic with the "test policy" command. In order to use this function, a policy must exist in the router configuration and the routes to be tested must exist in the router's routing table. This command is useful when first configuring policies, but is rarely utilized in more advanced configurations given that all routes to be tested must exist in the routing table. Below is an example of a typical usage of this feature.

```
      [edit routing-options static]
      jncia@my.router1# show
      route 10.192.0.0/10 discard;;
      route 10.224.0.0/11 discard;
      route 10.192.0.0/23 discard;
      route 10.192.0.0/24 discard;

      [edit policy-options]
      jncia@my.router1# show
      policy-statement testing-policy {
          term term1-route-filter {
              from {
                  route-filter 10.192.0.0/10 upto /14
      reject;
                  route-filter 10.192.0.0/16
      through /18;
                  route-filter 10.192.0.0/19
      orlonger;
              }
              then accept;
          }
      }
```

```
      jncia@my.router1> test policy testing-policy
10.192.0.0/10
      inet.0: 4 destinations, 4 routes (4 active, 0
holddown, 0 hidden)
      Prefixes passing policy:
      10.192.0.0/23 *[Static/5] 00:12:05
                              Discard
      10.192.0.0/24 *[Static/5] 00:12:05
                              Discard

      Policy test: 2 prefixes accepted, 2 prefixes
rejected
```

To better determine the effects of policy on routing updates or advertisements once policy is applied, output results from "`show route protocol` *`protocol`*" can be compared with those from "`show route advertising-protocol` *`protocol`*" to view the effects of export policy. Comparing the results with "`show route receive-protocol` *`protocol`*" will indicate the effects of the import policy.

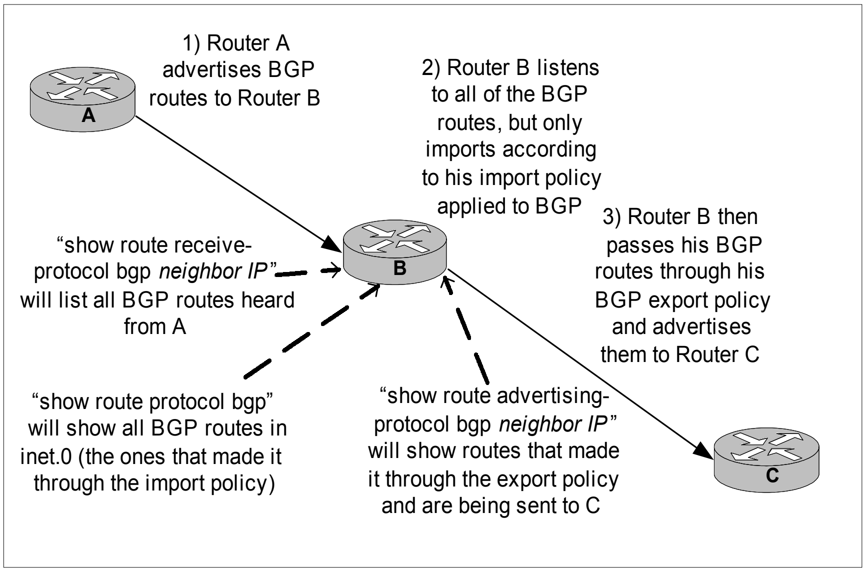

1) Router A advertises BGP routes to Router B
2) Router B listens to all of the BGP routes, but only imports according to his import policy applied to BGP
3) Router B then passes his BGP routes through his BGP export policy and advertises them to Router C
A
B
C
"show route receive-protocol bgp *neighbor IP*" will list all BGP routes heard from A
"show route protocol bgp" will show all BGP routes in inet.0 (the ones that made it through the import policy)
"show route advertising-protocol bgp *neighbor IP*" will show routes that made it through the export policy and are being sent to C

Figure 10.3 Seeing the results of routing policy

Firewall

The firewall within JUNOS is simply the application of policy to control packets to/from an interface. Apply a firewall filter 'input' on an interface, and it will analyze all traffic received on that interface. Conversely, applied on the 'output', it will take affect on transmitted traffic from the interface applied. If applied to loopback0, the filter will affect all traffic to/from the routing engine. Juniper routers with the Internet Processor II can perform advanced firewall filtering features. Note that only early M40 routers shipped with the Internet Processor I, which has a limited firewall filtering feature-set. You can apply no more than one firewall filter to each logical interface. Below is a common application of a firewall with key points highlighted:

```
[edit interfaces]
jncia@my.router1# show
    lo0 {
        unit 0 {
            description "logical router loopback0";
            family inet {
                filter {
                input all-needed-traffic-to-loopback0;
```

#<< traffic inbound to lo0 is subject to this firewall filter>

```
      }
      address 127.0.0.1/32;
      address 10.1.2.5/32 {
      preferred;
      }
```

Above, the firewall filter is applied to the logical interface lo0.0 under `[edit interfaces]`. Below, the configuration of the filter is shown at the `[edit firewall]` hierarchy level.

```
[edit firewall]
jncia@my.router1> show

  filter all-needed-traffic-to-loopback0 {
    term ospf-neighbors{
      from {
        source-address {
          10.4.8.0/24;
          192.18.18.0/24;
        }
        protocol ospf;
      }
      then {
        accept;
      }
    }
    term bgp-neighbors {
      from {
        source-address {
          10.0.0.0/25;
          192.168.1.0/24;
        }
        protocol tcp;
        port bgp;
      }
      then {
       accept;
      }
    }
    term trusted-management-hosts {
      from {
        source-address {
         10.0.0.0/8;
        }
      }
      then {
```

```
                accept;
            }
        }
        term deny-the-rest {   #<< One last term to deny
                                    anything else
            then discard;
        }
    }
```

The first term allows OSPF packets from the address blocks 10.4.8.0/24 and 192.18.18.0/24 to be accepted. The second term similarly allows packets from 10.0.0.0/25 and 192.168.0.0/24 using TCP to communicate to BGP. The third term allows packets coming from the trusted subnet 10.0.0.0/8 to be accepted. The final term is a catch all to ensure all other unauthorized packets are discarded. As can be seen, there is no *from* statement, so everything reaching this term is considered to match.

Policing

Within JUNOS, the ability to rate limit the amount of traffic into or out of an interface is known as policing. The most common use of this function is to control DOS and other types of attacks on the Internet. However, it can also be used to provide some degree of Quality of Service (QoS) on individual interfaces that are oversubscribed.

Policer statements are most often defined by themselves and then applied to interfaces or referenced in a firewall filter. The other option is to define each policer in each firewall in which it is used. Note that the policer must come before any term definitions.

Policy Configuration within JUNOS

Each policy begins under the configuration level 'policy-options' with a policy name. This name must be unique within the configuration on the router.

```
jncia@my.router1> show configuration policy-options
policy-statement example-policy {
        term bgp-peer-loopback {
            from {
                source-address {
                    127.0.0.1/32;
                }
            }
            then {
                accept;
            }
        }
```

As noted, each policy consists of one or more named terms. The term name is optional if only one term exists, but if used, must be unique within each policy.

```
jncia@my.router1> show configuration policy-options
policy-statement example-policy {
        term bgp-peer-loopback {
            from {
                source-address {
                    127.0.0.1/32;
                }
            }
            then {
                accept;
            }
        }
```

Each term can consist of at least one statement (‘from’ or ’to’) for match conditions. Any match statements are optional. If they are removed, all routes are considered to match. If multiple criteria are defined under the ‘from’ segment of a term, they must all be met to satisfy as a match for that term.

```
jncia@my.router1> show configuration policy-options
policy-statement example-policy {
        term bgp-peer-loopback {
            from {
                source-address {
                    127.0.0.1/32;
                }
            }
            then {
                accept;
            }
        }
```

Each term can contain a ‘then’ statement (optional). This statement determines the action taken on a route that has matched the ‘from’ or ‘to’ operators. There are three types of actions that can be taken on a matched route:

- “Flow control” actions include *accept*, *reject*, *next-term*, or *next-policy.*
- “Tracing options” report route matches to a specified log file.
- Actions that set properties associated with the route (MED, localpref, etc.).

If no ‘then’ statement is used, one of the following actions is triggered:

- If no flow control action is taken, the next term is evaluated.
- If there are no more terms, the next policy is evaluated.
- If there are no more terms or policies, the default action is taken (refer to figure 10.2).

```
jncia@my.router1> show configuration policy-options
policy-statement example-policy {
        term bgp-peer-loopback {
            from {
                source-address {
                    127.0.0.1/32;
                }
            }
            then {
                accept;
            }
        }
```

Key Points

As noted, JUNOS policy can range from a simple import policy to advanced rate-limiting after the application of firewall filters. Policy plays a key role in the operation of JUNOS routers, and for that reason it is addressed numerous times on the JNCIA exam. Read and understand all of the concepts within this chapter before test time for the best results.

- Each policy is named with a policy-statement
- Policies contain four key elements
  - Terms
  - From
  - Then
  - Action
- Each protocol has its own default policy for importing and exporting routes (see figure 10.2 for details on each)
- Firewalls are similar to JUNOS policies but are applied to interfaces instead of routing protocols
  - One firewall filter can be applied to each logical interface (one inbound and one outbound)
- Policing can be used to rate-limit traffic on interfaces

Appendix A: Sample Quiz

Chapter 1 Basics

1) How many bits represent the host portion of the IP address 192.168.10.64/27?

2) Expand the IP address and subnet mask 10.154.63.24/19 into their binary equivalents.

3) What are the network and broadcast addresses of the subnet including host 10.23.32.100/23?

4) How many useable host addresses can be configured in the subnet 192.168.45.0/28? What are the first and last useable host addresses?

5) A subnet requires addresses for 42 unique hosts. What is the smallest subnet that can be used to allocate enough addresses?

6) How many total addresses are in 192.168.10.0/25 that are not in 192.168.10.0/26?

7) Which address "class" does 64.123.123.100 belong to?
   a. Class A
   b. Class B
   c. Class C
   d. Class D

Chapter 2 Hardware

1) What is the name of the 100MB link between the PFE and the RE?
   a) So0/0
   b) Eth0
   c) Fxp0
   d) Fxp1

2) Which routers contain Packet Director ASICs? (choose all that apply)
    a) M160
    b) M40e
    c) M40
    d) M20
    e) M10

3) Which two components comprise the Host Module?
    a) PFE and RE
    b) RE and MCS
    c) PCG and MCS
    d) MCS and PFE

4) On which ASIC(s) does queuing take place?
    a) PIC ASIC
    b) I/O Manager ASIC
    c) DBM ASIC
    d) Internet Processor II

5) How many FPC slots are on an M40?
    a) 2
    b) 4
    c) 6
    d) 8

6) Which ASIC sends packets out a physical port?
    a) PIC ASIC
    b) I/O Manager ASIC
    c) DBM ASIC
    d) Internet Processor II

7) The FPCs are built into the FEB on which two platforms?
    a) M5
    b) M10
    c) M20
    d) M40
    e) M160

8) The M20 supports redundant Routing Engines.
    a) True
    b) False

9) What are the primary responsibilities of the RE?
    a) Control routing protocol traffic, perform route-lookups.
    b) Forward data traffic, perform route filtering.
    c) Maintain routing protocols, control software processes.
    d) Manage interfaces, reassemble packets from shared memory.

10) Match the following hardware components to the correct platform.

a)	M160	SSB
b)	M40	FEB
c)	M20	SFM
d)	M10	SCB

Chapter 3 JUNOS:

1) What is the standard boot sequence for JUNOS?
    a) PCMCIA flash, compact flash, hard-drive, network
    b) Compact flash, PCMCIA flash, network, hard-drive
    c) Hard-drive, compact flash, network, PCMCIA flash
    d) PCMCIA flash, compact flash, network, hard-drive

2) What is the route preference of a static route?
    a) 1
    b) 5
    c) 15
    d) 50

3) If there is a route known via RIP and OSPF (internal) which one will be installed (assuming the exact same route with no attributes altered)?
    a) RIP
    b) OSPF
    c) Neither since there is a tie.
    d) Both, they load balance.
    e) Whichever originated closer.

4) Into which table are static routes installed?
    a) inet.0
    b) inet.1
    c) inet.2
    d) inet.3

5) The forwarding table actually used to make next-hop forwarding decisions is stored where?
    a) The hard drive
    b) Compact flash
    c) The RE
    d) The PFE

6) Under which configuration statement is the BGP autonomous system number set?
    a) [edit policy-options]
    b) [edit protocols bgp]
    c) [edit routing-options]
    d) [edit system]

7) Which routing table contains MPLS information?
    a) inet.0
    b) inet.1
    c) inet.2
    d) inet.3

8) What is the CLI command to view the *messages* logfile?
    a) show logfile messages
    b) show messages
    c) show log messages
    d) monitor logfile messages

9) Explain the differences between routes held in the forwarding table and those in the routing table.

10) Under which configuration level is the Router ID set?
    a) [edit protocols]
    b) [edit router-options]
    c) [edit system]
    d) [edit routing-options]

11) Which two of the below are traits of JUNOS?
    a) It is based on a UNIX kernel
    b) There is specific code for each M-series platform
    c) Processes run independently
    d) The code can be updated without a service impact
    e) Internet Processor II

12) Which command will erase the current configuration and load the entire contents of “newconfig” from edit mode?
    e) Load replace newconfig
    f) Load merge newconfig
    g) Load override newconfig
    h) Load newconfig

13) Which command will configure an IP address upon fe-0/0/0.0from the [edit interfaces fe-0/0/0 unit 0] prompt?
    a) Set address 10.45.123.32/30
    b) Set family inet address 10.45.123.32/30
    c) Set address family inet 10.45.123.32/30
    d) Set inet family address 10.45.123.32/30

14) What are legal completions for a configured static route? (Choose all that apply)
    a) An interface
    b) An address
    c) Accept
    d) Discard
    e) Reject

15) Where is the backup copy of JUNOS kept?

Chapter 4 RIP

1) What is the maximum hop-count for a **reachable** RIP route?
    a) 15
    b) 16
    c) 10
    d) 255

2) What two mechanisms does RIP use to prevent routing loops (select 2)?
    a) Split-horizon
    b) Link-state database
    c) Random routing database checks
    d) Poison-reverse

3) In which routing table are RIP routes placed?

a) inet.0
b) inet.1
c) inet.2
d) inet.3

4) A RIP update, using authentication, can contain up to how many networks?

a) 1
b) 24
c) 25
d) 255

5) What type(s) of authentication does JUNOS support for RIPv1?

a) Plain-text password
b) MD5 encrypted password
c) Both a and c
d) Neither of these methods

6) What type(s) of authentication does JUNOS support for RIPv2?

a) Plain-text password
b) MD5 encrypted password
c) Both a and b
d) Neither of these methods

7) Define how split-horizon works to control routing loops?

8) Define how poison-reverse functions to control routing loops?

9) How do RIPv2 routers send route updates to their neighbors?

a) Unicast
b) Multicast
c) Broadcast
d) Labelcast

10) Which of the following is NOT a field in a RIPv2 update message?

a) Network address
b) Metric
c) Cost
d) Next-hop

Chapter 5 OSPF

1) What is the purpose of an ASBR?

2) What is the purpose of an ABR?

3) What type of OSPF network utilizes a DR?
    a) Point-to-point (P2P)
    b) NBMA
    c) Point-to-multi-point (P2MP)
    d) BMA

4) What address does the RID default to?

5) What type of router sends Type 2 Network LSAs on a broadcast segment?
    a) ABR
    b) ASBR
    c) DR
    d) BDR
    e) Drother

6) What type of router cannot exist in a stub area?
    a) ASBR
    b) ABR
    c) DR
    d) BDR
    e) Drother

7) Give 2 purposes of the Hello message.
    a) Exchange route information.
    b) Discover neighbors on a network segment.
    c) Maintain neighbor adjacencies.
    d) Discover the shortest path between a source and destination.

8) Which State is a router in after a Hello is sent but before one is received?
    a) Down
    b) Init
    c) 2way
    d) Drother
    e) Exchange
    f) Full

9) What are the default OSPF timer values?

10) Which OSPF router distributes area routes into other areas?
   a) ASBR
   b) ABR
   c) DR
   d) Level 1

11) Which CLI command will show the state of OSPF to other routers?
   a) show ospf interface
   b) show ospf adjacency
   c) show ospf neighbor
   d) show ospf detail

12) Which CLI command with show the type of networks the router participates in (Point to Point, BMA, etc)?
   a) show ospf interface
   b) show ospf adjacency
   c) show ospf neighbor
   d) show ospf detail

13) What is the CLI command to view the OSPF link-state database?

14) On an Ethernet segment Router A (priority 100), Router B and Router C (both priority 90) come online. Which one is elected to be the designated router?

15) On the same segment above in 14, Router D comes online 30 minutes later with a priority of 110. Assuming nothing else has changed, which router is now DR?

16) What is the minimum configuration necessary to run OSPF on a router?

17) What is the first State when establishing an OSPF adjacency?
   a) Down
   b) Init
   c) 2way
   d) Drother
   e) Exchange
   f) Full

18) In which State should routers be with their neighbors on a BMA network where neither are DR or BDR?
   a) Down
   b) Init
   c) 2way
   d) Drother
   e) Exchange
   f) Full

19) What State should synchronized neighbors on a point to point segment be in?
   a) Down
   b) Init
   c) 2way
   d) Drother
   e) Exchange
   f) Full

Chapter 6 IS-IS

1) The first byte of the Area ID is also known as:
   a) NSET
   b) N-selector
   c) DIS
   d) AFI

2) When a Level 1 router encounters packets destined for a different Area, what does it do?
   a) Forwards it off to the other Area itself.
   b) Sends it to the nearest L1/L2 router.
   c) Drops the packet and sends a "Host Unreachable" error back to the sender.
   d) Silently discards the packet and sends no error back to the sender.

3) What is the command to view which interfaces are configured for IS-IS?
   a) show is-is interface
   b) show protocol is-is interface
   c) show interface
   d) Show interface is-is

4) By default, an IS-IS enabled interface is placed into which Level?
    a) Level 1
    b) Level 2
    c) Level 3
    d) Both Level 1 and Level 2
    e) None, the Level must be manually enabled

5) What is the command to view the IS-IS database?

6) When configuring a router for IS-IS, the last byte of the NSAP must be set to what?

7) Three IS-IS routers (A, B, C) are brought online in an Ethernet segment. A has no priority configuration (default). B has priority set to 0. C has a priority set to 100. Who will be elected the DIS?
    a) Router A
    b) Router B
    c) Router C
    d) None. There is no such thing as a DIS for an Ethernet segment

8) Which of the following is/are functions of a PSNP (choose all that apply):
    a) Request a missing LSP
    b) Share routing table information
    c) Notify to retransmit an LSP after a sequence error is noted
    d) Reset the Hello timer

9) Which is better thought of as a “site” router (meaning no direct communication to the network backbone)?
    a) Level 1 router
    b) Level 1/2 router
    c) Level 2 router
    d) Level 3 router

10) Identify the portions of the NSAP address 49.0001.9aff.00ab.0030.1223.4045.00.

11) Which PDU is used to advertise a complete listing of all IS-IS LSPs in the router link state database?
    a) CSNP
    b) PSNP
    c) IIH
    d) LSP

12) If an IS-IS router detects a fault with its link state database, which PDU does it use to request an update?
    a) CSNP
    b) PSNP
    c) IIH
    d) LSP

Chapter 7 BGP

1) At what level of the configuration hierarchy is the AS number set?
    a) [edit protocols bgp]
    b) [edit system as]
    c) [edit routing-options]
    d) [edit protocols as]

2) What do the numbers in the final column of the blow output mean for the bgp neighbor? (active/received/dampened)

```
10.244.2.154 65333 452 432 0 4 5d19h 825/1263/0
```

    a) Routes received/advertised/dampened
    b) Routes advertised/received/dampened
    c) Routes received/active/dampened
    d) Routes active/received/dampened

3) How do you get BGP to advertise all OSPF routes to all neighbors?
    a) Export policy in OSPF that matches on BGP routes.
    b) Import policy in OSPF that matches on BGP routes.
    c) Export policy in BGP that matches on OSPF routes.
    d) Import policy in BGP that matches on OSPF routes.

4) Given equal cost BGP routes in inet.0 and inet.3, which will be selected?
    a) Neither
    b) Inet.0
    c) Inet.3
    d) random

5) Given the list of criteria for BGP route selection within JUNOS, put them in order.

    MED
    IGP cost
    AS-path length
    Local pref
    eBGP vs. iBGP
    origin code

6) Which type of BGP message contains information on new and changed routes?
    a) Update
    b) Flash
    c) Synch
    d) NLRI

7) Which port does BGP listen to?
    a) TCP 79
    b) TCP 179
    c) UDP 79
    d) UDP 179

8) What is local preference used for?

9) By default, BGP will advertise what routes to neighbors? (choose all that apply) A:(direct and BGPlearned)
    a) Directly connected routes
    b) Routes learned via RIP
    c) Routes learned via OSPF
    d) Routes learned via BGP

10) Which command is used to view specific BGP neighbor information?
    a) show bgp summary
    b) show bgp neighbor
    c) show bgp interface
    d) show bgp adjacency

11) iBGP peers are those that are:
    a) In the same AS.
    b) In different AS.
    c) Directly connected.
    d) Configured with the same address.

12) Unless BGP multi-hop is enabled, external BGP neighbors must be:
    a) In the same AS.
    b) In different AS.
    c) Directly connected.
    d) Configured with the same address.

13) If two or more valid paths are available, BGP will inherently:
    a) Per-packet load share.
    b) Per-flow load share.
    c) Per-prefix load share.
    d) Not load share.

14) iBGP neighbors must be directly connected.
    a) True
    b) False

15) A router configured with BGP can automatically discover BGP neighbors.
    a) True
    b) False

16) BGP split horizon dictates that:
    a) Routes learned via iBGP are not advertised back out the interface they were learned.
    b) Routes learned via iBGP are advertised to all BGP neighbors.
    c) Routes learned via iBGP are not advertised to any BGP neighbors.
    d) Routes learned via iBGP are not advertised to any iBGP neighbors.

17) Which two of the below help reduce the problems of requiring a full mesh of iBGP neighbors?
    a) Confederations
    b) BGP split horizon
    c) BGP multi-hop
    d) Route reflectors

18) How does BGP avoid routing loops? (choose two)
    a) iBGP consults its link state database upon learning routes and drops incorrect prefixes.
    b) BGP split horizon ensures routes do not propagate past immediate neighbors.
    c) eBGP checks the as-path attribute and drops prefixes that include its native AS.
    d) Only authenticated eBGP neighbors are allowed to advertise usable routes.

19) What can a MED from AS 65000 communicate to a BGP peer in AS 64666?
    a) Which interface or session would be preferred for AS 6500 to receive traffic.
    b) Which interface or session AS 6500 will be sending traffic on.
    c) Where and how many exit points there are between the two AS.
    d) How AS 64666 should route the packet internally.

Chapter 8 MPLS

1) What is an LSP?
    a) A unidirectional path through an internal network.
    b) A bi-directional path through an internal network.
    c) A traffic-engineered path.
    d) A pointer for IP forwarding.

2) Which of the below are valid label operations?
    a) Pop
    b) Drop
    c) Swap
    d) Push
    e) Attach
    f) Delete

3) What object specifies the next-hop of an LSP?
    a) RRO
    b) ERO
    c) TED
    d) CSPF

4) What are the advantages of signaled versus static LSPs? (choose all that apply)
    a) Dynamic LSP creation.
    b) Less administrative overhead.
    c) Less latency.
    d) Easier QoS.

5) What functions does a router perform when receiving an MPLS packet with a single label 0?
    a) Drop the packet silently
    b) Drop the packet and return an "ICMP unreachable" message.
    c) Pop the label and forward it to the RE.
    d) Pop the label and forward to the IP destination.

6) MPLS does not require an underlying routing protocol to function.
    a) True
    b) False

7) Is it possible to use an LSP as a BGP next hop?
    a) True
    b) False

8) Name two MPLS protocols for signaling LSPs.
    a) RSVP
    b) OSPF
    c) LDP
    d) TCP

9) An LSP is signaled from Router A through Router B and C to Router D. Which is the penultimate hop?
    a) Router A
    b) Router B
    c) Router C
    d) Router D

10) An LSP is configured with strict constraints. The ingress router sees the ERO with a strict constraint of next hop 10.1.2.4. What must be true of this address?
    a) Must not be in the LSP path
    b) Must be in the LSP path at some point
    c) Must be directly connected to previous LSP hop.
    d) It must be the ingress router.

11) What is contained in the RRO?
    a) A list of all hops within an LSP.
    b) A list of all labels within an LSP.
    c) A list of all constraints for an LSP.
    d) A list of all LSPs.

12) What is the primary purpose of the RRO?
    a) To determine labels for an LSP.
    b) To prevent loops within an LSP.
    c) To allocate bandwidth for an LSP.
    d) To list all available, reachable LSPs.

13) What are the two ways a router can allocate labelspace?
    a) Per-interface
    b) Global
    c) Per-router
    d) Arbitrary

14) What functions does a router perform when receiving an MPLS packet with top label 1?
    a) Drop the packet silently
    b) Drop the packet and return an “ICMP unreachable” message.
    c) Pop the label and forward it to the RE.
    d) Pop the label and forward to the IP destination.

15) If an LSP is configured with a secondary path and 'standby' enabled, when is the secondary path signaled?
    a) When the router comes up.
    b) When the primary fails.
    c) At the same time the primary is signaled.
    d) As soon as RSVP is configured.

16) When an LDP router receives multiple label bindings for a single destination what metric is used to select the best label?
    a) IGP metric
    b) LDP metric
    c) Local Pref
    d) Route preference

17) What information is used to compile the TED? (select all that apply)
    a) BGP NLRI.
    b) OSPF type-10 LSAs.
    c) Protocol reachability information obtained from CSPF.
    d) IS-IS TLV extensions.

18) What happens if an MPLS router receives an IP packet destined for a network for which the router has no label, but has a standard IP route?
    a) The packet is discarded.
    b) The packet is forwarded via standard IP forwarding.
    c) The packet is rejected and an ICMP message is sent back to the sending host.
    d) The router requests a label from the egress router.

19) What information is stored in the inet.3 routing table?
    a) BGP routes
    b) OSPF routes
    c) LSPs
    d) Labels

Chapter 9 Multicast

1) What is the range of available IP Multicast addresses?
    a) 224.0.0.0 – 244.255.255.255
    b) 222.0.0.0 – 244.0.0.0
    c) 224.0.0.0 – 239.255.255.255
    d) 224.0.0.0 – 239.0.0.0

2) What is the purpose of IGMP?
    a) To communicate the SPT back to the multicast source.
    b) To control the hosts that join and leave multicast streams.
    c) To discover the shortest path back to the RP.
    d) To forward multicast traffic from a host.

3) What is an advantage of IGMPv2 over IGMPv1?
    a) Version 2 has run a shortest path calculation to the multicast source.
    b) Version 2 allows for a single data stream, Version 1 requires individual feeds.

c) Version 2 uses explicit Leave messages to reduce unneeded traffic on the LAN.
d) Version 2 requires significantly lower administration.

4) What is the biggest advantage of Shortest Path Trees (SPTs) compared to Shared Trees (STs)?
   a) SPTs minimize bandwidth usage by distributing multicast streams from a common point.
   b) SPTs utilize less routing overhead when forwarding packets.
   c) SPTs minimize latency by finding the optimal path between each source and each receiver.
   d) SPTs require less state handling and memory overhead on multicast routers.

5) What is an advantage of using Shared Trees?
   a) Shared trees minimize bandwidth usage by distributing multicast streams from a common point.
   b) Shared Trees utilize less routing overhead when forwarding packets.
   c) Shared Trees minimize latency by finding the optimal path between each source and each receiver.
   d) Shared Trees require less state handling and memory overhead on multicast routers.

6) What information does the router use to do an RPF check?
   a) The unicast routing table
   b) The DVMRP table
   c) The PIM table
   d) The MSDP table

7) Why is Protocol Independent Multicast (PIM) called Independent?
   a) It is an open source, independently produced protocol.
   b) It works with any type of multicast traffic.
   c) It works with each multicast flow independently.
   d) It works with any IP unicast routing protocol

8) What is the main advantage of MBGP?
   a) It provides redundant unicast paths.
   b) It provides session redundancy.
   c) It provides redundant multicast paths.
   d) It allows for dissimilar unicast and multicast routing topologies.

9) What is the purpose of an (S, G) notation?
   a) To identify the source's unicast address with a multicast group address.
   b) To identify the source's multicast address with a unicast group address.
   c) To identify the source's unicast address with a unicast group address.
   d) To identify the source's multicast address with a multicast group address.

10) What two things do dense mode multicast protocols assume?
   a) That many hosts require the multicast stream.
   b) That the RP will begin to administer the source stream to interested groups.
   c) That bandwidth is plentiful.
   d) That no one will be pruned from the source stream.
   e) That end hosts are running PIM.

11) What is the purpose of MSDP?
   a) To exchange (S, G) information with RPs in other autonomous systems.
   b) To exchange RPF information with another AS
   c) To exchange multicast streams with non-IGMP enabled hosts.
   d) To determine if a received multicast stream is valid.

12) What is the purpose of MBGP?
   a) To exchange (S, G) information with another AS.
   b) To exchange RPF information with another AS
   c) To exchange multicast streams with non-IGMP enabled hosts.
   d) To determine if a received multicast stream is valid.

13) Which of the below things do sparse mode multicast protocols assume? (choose two)
   a) That many hosts require the multicast stream.
   b) That the RP will begin to administer the source stream to interested groups.
   c) That bandwidth is scarce.
   d) That no one will be pruned from the source stream.
   e) That end hosts are running PIM.

Chapter 10 Policy

1) How many firewall filters can be placed on an interface

a) One per physical
b) One per logical
c) One per logical in/out
d) One per physical in/out

2) Select two policy actions from the below:

a) Reject
b) Pop
c) Push
d) Accept

3) If three match conditions are specified within a single accept term, when will a route be accepted?

a) When it matches the first condition.
b) As soon as it matches a single condition.
c) When it matches all three conditions.
d) If it matches none of the conditions.

4) What is the difference between a prefix-list and a route-filter? (Choose all that apply)

a) A route filter can specify an immediate action upon matching
b) A prefix-list can specify an immediate action upon matching.
c) A prefix-list can specify address ranges
d) A route filter can specify address ranges

5) Routes learned from an eBGP neighbor are subject to what type of policy before entering the routing table?

a) export
b) firewall
c) inbound
d) import
e) outbound

6) Policy statements can be applied to which of the following?

a) CSPF
b) Routing Protocols
c) SFMs
d) Interfaces

7) If no accept or reject is explicitly configured for a BGP policy term what is the default behavior for importing routes?

8) If a policy statement is applied to a specific BGP peer and a different policy is applied to the group to which that peer belongs, which takes effect?
   a) Group
   b) Peer
   c) Both
   d) Neither

9) Within a policy the 'from' statements define which of the following?
   a) match conditions
   b) actions
   c) term
   d) End of policy

10) Which of the following policy-statement operators are used to change the attributes of a route?
   a) from
   b) then
   c) accept
   d) push

11) If a route doesn't match a given term the default behavior is to:
   a) Evaluate the route against the next term
   b) Evaluate the route against the next policy-statement
   c) Reject the route
   d) Accept the route

12) Which of the following are valid match criteria within a policy-statement? (choose all that apply)
   a) source address
   b) protocol
   c) as-path
   d) next-hop

13) "192.168.64.0/19 orlonger" matches:
   a) 192.168.64.0/18
   b) 192.168.64.0/19
   c) 192.168.64.0/20

14) “192.168.64.0/19 upto /24” matches which address block(s)?

a) 192.168.64.0/19
b) 192.168.64.0/20
c) 192.168.64.0/23
d) 192.168.64.0/24

Appendix B: Additional Information

Possible ‘show’ command completions in JUNOS 5.0:

```
accounting       Show accounting profiles
aps              Show APS information
arp              Show system ARP table entries
as-path          Show table of known AS paths
bgp              Show information about BGP
chassis          Show chassis information
cli              Show cli settings
configuration    Show configuration file contents
connections      Show CCC connections
dvmrp            Show information about DVMRP
firewall         Show firewall counters and info
host             Hostname lookup using DNS
igmp             Show information about IGMP
ilmi             Show ILMI information
interfaces       Show interface information
isis             Show information about IS-IS
l2vpn            Show information about L2VPNs
ldp              Show information about LDP
log              Show contents of a log file
mpls             Show information about MPLS
msdp             Show information about MSDP
multicast        Show multicast information
ntp              Network Time Protocol information
ospf             Show information about OSPF
pfe              Show packet forwarding engine data
pim              Show information about PIM
policy           Show policy information
rip              Show information about RIP
route            Show routing table information
rsvp             Show information about RSVP
sap              Session advertisement addresses
snmp             Show SNMP information
system           Show system information
task             Show protocol per-task information
ted              Show information about TED
```

```
version          Show software revision levels
vrrp             Show VRRP information
```

Possible '[edit] show' configuration completions in JUNOS 5.0:

```
accounting-options   Accounting data configuration
apply-groups         Groups to get config data from
chassis              Chassis configuration
class-of-service     Class-of-service configuration
firewall             Define firewall configuration
forwarding-options   Packet sampling options
groups               Configuration groups
interfaces           Interface configuration
policy-options       Routing policy options
protocols            Routing protocol configuration
routing-instances    Routing instance configuration
routing-options      Protocol-independent options
snmp                 Simple Network Mgmt Protocol
system               System parameters
```

Media types as abbreviated in interface names:

- `ae`—Aggregated Ethernet interface. A virtual bundled Ethernet link.
- `as`—Aggregated SONET/SDH interface. A virtual bundled SONET link.
- `at`—ATM interface.
- `ds`—DS-0 interface (on a channelized DS-3 or E1 PIC).
- `e1`—E1 interface
- `e3`—E3 interface
- `es`—Encryption interface
- `fe`—Fast Ethernet interface
- `fxp`—Management and internal Ethernet interfaces
- `ge`—Gigabit Ethernet interface
- `gr`—Generic Route Encapsulation (GRE) tunnel interface
- `ip`—IP-over-IP encapsulation tunnel interface
- `lo`—Loopback interface

- `ml`—Multilink interface
- `mo`—Passive monitoring interface
- `mt`—Multicast tunnel interface
- `so`—SONET/SDH interface
- `t1`—T1 interface (includes channelized DS-3 interfaces)
- `t3`—T3 interface (includes channelized OC-12 interfaces)
- `vt`—VPN loopback tunnel interface

Family types allowed on interfaces:

- Internet Protocol, version 4 (IPv4)
- Internet Protocol, version 6 (IPv6)
- International Organization for Standardization (ISO)
- Multiprotocol Label Switching (MPLS)
- Circuit cross-connect (CCC)
- Translational cross-connect (TCC)
- Multilink Frame Relay (MLFR)
- Multilink PPP (MLPPP)
- Trivial Network Protocol (TNP)

Well Known Multicast Addresses:

224.0.0.0	Base multicast address.
224.0.0.1	All local hosts multicast group. IGMP Query message
224.0.0.2	All routers.
224.0.0.4	DVMRP

224.0.0.5	All OSPF Routers address. Sends routing updates to all OSPF routers on a network segment.
224.0.0.6	OSPF DR address. Sends routing updates to OSPF Designated Router.
224.0.0.9	RIPv2
224.0.0.13	PIMv2

Appendix C: Glossary of Terms

ABR – Area Border Router. In OSPF, a router that has interfaces in a site area as well as Area 0. Traffic flowing from one area to another traverses an ABR.

Adjacency – When two routers running the same protocol are configured to talk to one another and exchange routing information, they can become adjacent. Also sometimes called Neighbors.

AFI – Address Family Identifier. The first byte of an NSAP Area ID which tells the system how to interpret the rest of the Area ID field. The Area ID is an AS local trait. The AFI byte is commonly set to 49.

Area – A logical grouping of devices configured in a routing protocol to either restrict administrative overhead and traffic.

AS – Autonomous System. A set of network devices under common administrative control. An IGP is considered to be the protocol used to reach nodes within the same AS. An EGP is used to reach destinations outside of the AS.

AS-path – An attribute carried by BGP routes that reflects the number of Autonomous Systems traffic will need to transit to reach the destination. AS path is one of the primary means of BGP route selection.

ASBR – Autonomous System Boundary Router. In OSPF a router that separates two different routing domains, i.e. a router that injects RIP routes into OSPF.

ASIC – Application Specific Integrated Circuit. A piece of hardware, specifically a chip, designed to perform a specific function. Functions carried out by hardware are conducted many times faster than those done via software processes. Juniper routers utilize a number of ASICs in the PFE.

ATM – Asynchronous Transfer Mode – A high-speed broadband method of transporting data between nodes using fixed 53 byte cells. Useful for applications that require rigid Quality of Service (QoS) and jitter tolerances such as voice and video.

Backbone – A group of routers that are responsible for ferrying traffic between different areas. Backbone routers are usually connected by high-speed circuits and do not normally directly serve end user hosts.

BGP – Border Gateway Protocol. A protocol for exchanging routes between autonomous systems (AS). BGP version 4 is the de facto EGP in use on the Internet.

Bit – A digital signal that can exist in one of two possible positions. These positions are often referred to as '1' and '0', or 'on' and 'off'.

BMA – Broadcast Multi Access. A type of LAN, such as Ethernet, where connected devices are capable of broadcasting to all other connected nodes. The ability for any node to broadcast repeatedly can quickly overwhelm normal data transmissions if considerations are not made to handle such events.

Broadcast – One-to-all traffic flow. This type of data transmission is not an economic use of network resources, as all hosts on a segment receive the traffic regardless of whether or not they will use it.

CIDR – Classless Interdomain Routing. A system of classless IP addressing where the subnet mask may be calculated at any bit across the address. This allows for more efficient use of address space. See also *VLSM.*

CIP – Connector Interface Panel. Located on the far left of the FPC card cage, the CIP contains the console, auxiliary, and management Ethernet connections for the router. Host 0 and Host 1 each have their own dedicated ports.

CLI – Command Line Interface. In JUNOS the prompt at which commands are issued to the router.

CoS – Class of Service. A way of managing network traffic by grouping types of data traffic together and defining a service class for each group. (For example, a voice group, e-mail group, file transfer group, telnet group, etc.) This differs from QoS which guarantees network resources for priority traffic. CoS is coarser and simpler to implement than QoS.

CSPF – Constrained Shortest Path First. The algorithm which is run by RSVP to determine the best path for an LSP. CSPF takes into account any hop requirements that may be placed on the LSP calculation.

CSNP – Complete Sequence Number PDU - Contains an entire listing of LSPs in the database. Sent out all IS-IS links periodically to ensure all routers have synchronized LSP databases. A DIS will multicast the CSNP on a broadcast network media to limit acknowledgement traffic.

Daemons – A process that runs in a UNIX environment. Daemons run independently of one another and are more resilient to failure.

Dead timer/Dead Interval – The amount of time a protocol will allow to pass without receiving a keepalive message before considering the neighbor to be unreachable. See also *keepalive*.

DIS – Designated Intermediate System. The node in an ISIS broadcast network segment appointed to propagate announcements to conserve overhead. Equivalent to a DR in OSPF, but there is no backup (BDR) in an ISIS implementation.

DR – Designated Router. 1) In OSPF the router responsible for keeping all its adjacent neighbors synchronized on a multi-access LAN segment. 2) In PIM on a multi-access LAN, the DR is responsible for sending Joins to the Rendezvous Point (RP) to construct the shared tree

DVMRP – Distance Vector Multicast Routing Protocol. A multicast routing protocol used to dynamically generate shortest path trees for forwarding multicast streams.

EGP – Exterior Gateway Protocol - A system of rules defining the procedure to route packets between different ASs. BGP is the most widespread EGP.

Egress Router – The device where MPLS frames exit an LSP after being label switched from another MPLS-enabled router.

ERO – Explicit Route Object. In MPLS the result of the best path calculation that is passed on to RSVP to be used for signaling the LSP.

FEC – Forward Equivalency Class. In MPLS the FEC defines which packets will be appended with a particular label. For example all packets with the same destination network address will be encapsulated with the same label before they are forwarded to the next router.

Forwarding Table – The table assembled on the RE and exported to the PFE which contains the selected best next-hops for all destinations. Smaller than the *routing-table*.

FPC – Flexible PIC Concentrator. On the larger M-series routers, FPCs hold up to 4 PICs in a chassis slot. The FPC contains the shared memory for all PICs in that slot as well as ASICs necessary for packet flow to and from the rest of the PFE.

Gateway – A legacy word used to describe routers. ISIS and OSI standards still can refer to routers in this way.

Hello protocol – Utilized by OSPF to establish and maintain neighbor adjacencies. Also used as a *keepalive* message.

IGMP – Internet Group Management Protocol. In multicast, the protocol that controls membership in a multicast group, allowing hosts to join and leave a stream.

IGP – Interior Gateway Protocol - A system of rules defining the procedure to route packets within a common AS. OSPF, RIP, an IS-IS are examples of an IGP.

IIH - IS-IS Hello PDU - Broadcast to determine neighboring IS-IS systems. Discovers and differentiates between Level 1 and Level 2 routers.

Ingress Router – The device where MPLS frames enter an LSP to be label switched to another MPLS-enabled router.

IP – Internet Protocol. Principle routed protocol on the Internet today, carrying all types of data through all types of media.

IP Address - A unique 4-byte number, normally expressed in four dotted decimal notation (eg 12.34.56.78), that identifies a specific host on a network. Certain IP addresses are reserved for special use and are not generally routed on the Internet.

IS-IS – Intermediate System to Intermediate System. A link state IGP designed for robust and complex networks. Utilizes OSI addressing.

JUNOS – The operating system written by Juniper Networks for its series of routers. Based on a UNIX kernel, it is robust and resilient.

Keepalive - A timer utilized by a routing protocol to ensure a neighbor is still reachable. If a keepalive message is not received within a configured interval, the neighbor will be considered unreachable and action will be taken upon the route topology. Also see *dead-interval*.

LAN – Local Area Network. A collection of nodes that are relatively close in physical location. Current networks usually depend on Ethernet at the LAN level.

LDP – Label Distribution Protocol. In MPLS a protocol describing the mechanism for exchanging, withdrawing and updating label information between neighboring routers.

LER – Label Edge Router. A router that sits at the edge of an MPLS network. It can either receive MPLS labeled packets, strip those headers and forward the IP packet within, or the reverse and send it off the another MPLS router.

Level 1 Intermediate System – In ISIS a node that routes packets only within its own Area. When dealing with packets destined for a different Area, the router sends them to the closest Level 2 node. A router running ISIS may have interfaces in Level1, Level 2, or both.

Level 2 Intermediate System – In ISIS a node that routes packets between and toward other Areas. A router running ISIS may have interfaces in Level1, Level 2, or both.

Local Preference - An attribute carried by BGP routes that reflects the administrative preference to select a route within the local AS. The number is configured through policy. A higher local-pref indicates a more desirable route. Local-pref is one of the primary means of BGP route selection.

Loopback – A logical router interface. Normally used as an endpoint for router-to-router communication or as a Router ID. Because a loopback is not tied to a physical interface it is more resilient to failure.

LSA – Link State Advertisement. In OSPF a protocol announcement of link state to other OSPF routers.

LSP – Label Switch Path. In MPLS, a unidirectional virtual path connecting two LSR/LER nodes through any number of intermediate MPLS active transit LSRs.

LSP – Link State PDU. In ISIS, the packets containing information regarding the state of the connections to adjacent routers.

LSR – Label Switch Router. A router that forwards packets based solely on label information. Transit LSP routers are LSRs.

MAC – Media Access Controller. Sometimes called the hardware address, the 6 byte (48 bit) MAC address is used to uniquely identify a network node on a LAN at layer-2 of the OSI model.

MBGP – Multiprotocol Border Gateway Protocol. Multicast protocol that allows autonomous systems to exchange information used for RPF checks.

MCS – Miscellaneous Control Subsystem. A hardware component in the M160/M40e platform that makes up part of the RE. The MCS provides monitoring and control of the router systems.

MED – Multi-Exit Discriminator. An route selection attribute in BGP. MEDs can be exchanged across eBGP sessions and tell the external neighbor which path is preferred. They usually reflect IGP cost, and as such a lower MED more preferable.

Metric – An administratively defined cost for taking a particular path through a network.

MPLS – Multi-Protocol Label Switching. A method of forwarding data packets based on a label rather than a longest match IP lookup. MPLS enables more stringent levels of packet delivery QoS than pure packet-switched networks.

MSDP – Multicast Source Distribution Protocol. A multicast protocol that allows RPs to exchange (S, G) groups.

Multicast – One-to-many traffic flow. Not as wasteful of bandwidth and resources as broadcasting.

NET – Network Entity Title. In ISIS a special NSAP that identifies the node rather than a physical interface. Can be thought of as a loopback address.

NSAP – Network Service Access Point. In ISIS the address used to identify a particular network connection, such as a router interface.

NSSA – Not so stubby area. An area in OSPF that allows a local ASBR to flood External LSAs within and out of the area, but still receives a default route for routes outside the area.

OSPF – Open Shortest Path First. A link state IGP designed for robust and complex internetworks. Written and designed for TCP/IP.

PCG – PFE Clock Generator. A hardware component in the M160/M40e platform responsible for synchronizing the internal PFE components.

PDU – Protocol Data Unit. The packets used to communicate between ISIS nodes. More generally, any packets used by a routing protocol to pass information.

PFE – Packet Forwarding Engine. The collection of hardware responsible for routing and switching data packets in a Juniper router. It includes the PICs, FPCs, and many ASICs in the M series.

PIC – Physical Interface Card. The hardware component where physical connection to the transport media is made. Come in a variety of speeds, but designed to be interchangeable across platforms.

PIM – Protocol Independent Multicast. A multicast protocol that functions independent of the IGP used. Determines pathing for source-destination traffic in a multicast traffic flow. Can be configured in sparse or dense mode.

Poison Reverse – In RIP, advertisements sent back to the originating router that contain a metric of 16. This marks that path as unreachable. This has the effect of preventing routing loops between nodes.

Port – A value from 1 to 65535 that is used to uniquely identify a TCP/IP application and allow communication between two hosts on a network. There is a set for TCP and another for UDP applications. A port, when combined with an IP address, form a “socket” through which applications can communicate.

PSNP - Partial Sequence Number PDU – Request for update PDU. When a CSNP receiving router detects a fault with its LSP table it sends a PSNP to the router that originally transmitted the CSNP requesting the missing LSP. The PNSP transmitting node is then forwarded the missing LSP.

QoS - Quality of Service. A method of controlling the delivery of data through a network by giving differing levels of priority to different traffic queues. Packets with a high QoS are given higher priority to network resources (e.g. bandwidth, processing) which allows for more consistent delivery.

Queuing – The process of giving priority to packets marked with high QoS when forwarding traffic. Queuing enables a packet switched network to emulate the stringent delivery requirements usually maintained by ATM and circuit switched networks.

RE – Routing Engine. The collection of hardware components responsible for control plane traffic in a Juniper. The RE sends and receives all routing protocol updates and is responsible for keeping the routing table up to date.

RIP – Routing Information Protocol. A distance vector routing protocol suitable for small networks.

Route – Information that describes the network path to a destination host.

Route Reflection – A system in BGP that allows for the propagation of routes learned from an iBGP neighbor to other iBGP neighbors, thereby eliminating the need for a large full-mesh of iBGP routers. Utilizes route-reflectors and route-reflector clients.

Routing Table – The database assembled by the RE from all the information provided by active routing protocols that describes paths to every known destination. The routing table contains every known path. Best next-hops are calculated and used to build the *forwarding table*.

RP – Rendezvous Point. A router configured to be the point at which the multicast source stream first locates the destination group.

RSVP – Reservation Protocol. A label distribution protocol for MPLS that enables routers to establish and maintain LSPs for forwarding packets. RSVP allows dynamic LSP creation based upon available bandwidth and administrator preferences.

SCB – System Control Board. A hardware component in the M40 platform that is part of the PFE. The SCB makes forwarding decisions, monitors the system, and controls the FPCs.

SFM – Switching and Forwarding Module. A hardware component in the M160/M40e platform that is part of the PFE. SFMs make forwarding decisions and distribute packets to shared memory.

Split Horizon – In BGP, the process of an not propagating iBGP updates past a single iBGP neighbor. EBGP neighbors will still receive updates. Route reflection allows for iBGP neighbors to supercede this rule when passing advertisements to cluster clients.

Split Horizon – In RIP, the process of not sending advertisements back to the router from which they were received. This prevents the routers from creating a loop where each thinks the other has the valid path.

Static route – Manually configured routing information. Static routes require administrator intervention to change any information, and as such cannot respond to changes in topology or traffic.

Stub Area – A site area in OSPF that receives a default route to reach prefixes outside it's own. By default, a stub area cannot contain an ASBR or a *virtual link*.

TCP – Transmission Control Protocol. A connection oriented protocol that runs on top of IP. TCP contains methods of ensuring all packets sent by the source are received by the destination host. Contrast with *UDP*.

TED – Traffic Engineering Database. For MPLS a table populated with information from the IGP that is used by CSPF to calculate the best path for an LSP.

Traffic Engineering – The process of tailoring traffic flows between nodes for optimum performance. RSVP allows traffic engineering based on bandwidth availability and static tie downs.

UDP – User Datagram Protocol. A connectionless protocol that runs on top of IP. UDP/IP has no method of error recovery and does not guarantee transport of packets, relying on higher level protocols to compensate. Contrast with *TCP*.

Unicast – One-to-one traffic. A single server and single client exchange data by means of network unique addresses. Unicast traffic flows do not scale well in terms of bandwidth used as the number of clients increase. Each new client requires a separate and dedicated flow, leading to rapid congestion for popular, high bandwidth applications. Multicast is seen as the solution to this problem. The majority of traffic flowing on the Internet today is unicast.

VLSM – Variable Length Subnet Mask. Allows for the custom configuration of the subnet to as many, or as few hosts as the administrator requires by defining the mask anywhere along the 32 bits of an IP address rather than at the traditional octet breaks of bits 8, 16, and 24. VLSM are usually noted as a slash followed by the number of bits in the network portion. (192.168.1.1/27) *See also CIDR*.

VPN – Virtual Private Network. A network with access limited to specific hosts tunneled over a wide, public access network (e.g. the Internet) and security is provided by encryption and/or special protocols.

WAN – Wide Area Network. Network connecting physically remote locations, usually comprised of high-speed leased lines.

Appendix D: Quiz Answers

Chapter 1: Basics

1) How many bits represent the host portion of the IP address 192.168.10.64/27?

Answer: 5. There are 32 bits in an IPv4 address. The subnet mask denotes the number of bits in the network segment, and the remainders describe the host portion. A VLSM /27 leaves 5 bits.

2) Expand the IP address and subnet mask 10.154.63.24/19 into their binary equivalents.
    10.154.63.24 = 00001010. 10011010. 00111111. 00011000
    255.255.224.0 = 11111111.11111111. 11100000.00000000

3) What are the network and broadcast addresses of the subnet including host 10.23.32.100/23?

The network address is the host portion set to all zeros, or 10.23.32.0 in this instance. The broadcast address is the host portion set to all ones, which is 10.23.33.255 in this case.

4) How many useable host addresses can be configured in the subnet 192.168.45.0/28? What are the first and last useable host addresses?

14 useable host addresses, the first being 192.168.45.1 the last 192.168.45.15. A /28 subnet leaves 4 bits for the host segment. Four bits allows for 16 combinations (2^4). Subtract one address for the broadcast and one for the network, leaving 14 total available addresses (2^n-2).

5) A subnet requires addresses for 42 unique hosts. What is the smallest subnet that can be used to allocate enough addresses?

Answer: /26. A /26 allows for 62 unique hosts. The subnets to either side allocate too many (/25 = 126 hosts) or too few (/27=30 hosts)

6) How many total addresses are in 192.168.10.0/25 that are not in 192.168.10.0/26?

Answer: 64. A /25 contains 128 total addresses, while the smaller /26 contains 64. This leaves 64 addresses that do not overlap.

7) Which address "class" does 64.123.123.100 belong to?
a) Class A. The first bit is 0, indicating a Class A address.

Chapter 2 Hardware

1) What is the name of the 100Mb link between the PFE and the RE?
d) Fxp1. Note: Fxp0 is the management Ethernet connection.

2) Which routers contain Packet Director ASICs? (choose all that apply)
a) M160
b) M40e

3) Which two components comprise the Host Module?
b) RE and MCS
These are two separate parts, but they must function together as one logical unit.

4) On which ASIC(s) does queuing take place?
b) I/O Manager ASIC

5) How many FPC slots are on an M40?
d) 8. Numbering begins at 0 and ends with 7.

6) Which ASIC sends packets out a physical port?
a) PIC ASIC

7) The FPCs are built into the FEB on which two platforms?
a) M5
b) M10

8) The M20 supports redundant Routing Engines.
a) True

9) What are the primary responsibilities of the RE?
c) Maintain routing protocols, control software processes

10) Match the following hardware components to the correct platform:

a) M160 =	SFM
b) M40 =	SCB
c) M20 =	SSB
d) M10 =	FEB

Chapter 3 JUNOS

1) What is the standard boot sequence for JUNOS?
   a) PCMCIA flash, compact flash, hard-drive, network

2) What is the default route preference of a static route?
   b) 5

3) If there is a route known via RIP and OSPF (internal), which one will be installed (assuming the exact same route with no attributes altered)?
   b) OSPF
   OSPF internal has a default protocol preference of 10, where RIP is 100 (lower preference wins)

4) Into which table are static IPv4 routes installed?
   a) inet.0

   inet.0 - Default unicast table
   inet.1 - Default multicast table
   inet.2 – Multicast RPF checks
   inet.3 - MPLS path information

5) The forwarding table actually used to make next-hop forwarding decisions is stored where?
   d) the PFE

6) Under which configuration statement is the BGP autonomous system number set?
   c) [edit routing-options]

7) Which routing table contains MPLS information?
   d) inet.3

8) What is the CLI command to view the *messages* logfile?
   c) show log messages
   To view any traceoption log files, simply type 'show log *filename*'

9) Explain the differences between routes held in the forwarding table and those in the routing table.

    The routing table includes all possible paths to destinations learned from every protocol. Routes in the forwarding table are the **best** next-hop path. The information in the forwarding table is used to make forwarding decisions.

10) Under which configuration level is the Router ID set?

    d) [edit routing-options]

    Both the Router ID and autonomous system number are set at this level of the configuration.

11) Which two of the below are traits of JUNOS?

    a) It is based on a UNIX kernel

    c) Processes run independently

12) Which command will erase the current configuration and load the entire contents of "newconfig" from edit mode?

    c) Load override newconfig

13) Which command will configure an IP address upon fe-0/0/0.0 from the [edit interfaces fe-0/0/0 unit 0] prompt?

    b) Set family inet address 10.45.123.32/30

14) What are legal completions for a configured static route? (Choose all that apply)

    a)An interface

    b)An address

    d)Discard

    e)Reject

15) Where is the backup copy of JUNOS kept?

    Answer: The hard drive.

Chapter 4 RIP

1) What is the maximum hop-count for a **reachable** RIP route?

    a) 15

    A hop count of 16 indicates an unreachable route. This value was not changed for version 2, limiting the scalability of RIP in larger networks.

2) What two mechanisms does RIP use to prevent routing loops (select 2)?

a) Split-horizon
d) Poison-reverse

3) In which routing table are RIP routes placed?

a) inet.0
inet.0 - Default unicast table
inet.1 - Default multicast table
inet.2 – Multicast RPF checks
inet.3 - MPLS path information

4) A RIP update, using authentication, can contain up to how many networks?

b) 24

25 updates can be carried in a RIP update message. Only 24 can be carried if authentication is used.

5) What type(s) of authentication does JUNOS support for RIPv1?

d) Neither of these methods

Authentication was not introduced until RIPv2

6) What type(s) of authentication does JUNOS support for RIPv2?

c) Both a and b

The RFC for version 2 specifies plain-text passwords, but JUNOS allows plain-text and MD5 encrypted keys.

7) Define how split-horizon works to control routing loops?

Answer: A route learned through an interface will be forwarded out all interfaces EXCEPT that interface through which it was learned.

8) Define how poison-reverse functions to control routing loops?

When a route is learned through an interface, the same route is sent back through that interface with an unreachable metric (16 in the case of RIP), thus preventing a loop.

9) How do RIPv2 routers send route updates to their neighbors?

b) Multicast

RIPv1 broadcasts messages. RIPv2 is more advanced and uses multicast to a predefined address.

10) Which of the following is NOT a field in a RIPv2 update message?

c) Cost

RIP has no concept of cost. It uses hop count as the metric to choose the best route to a destination.

Chapter 5 OSPF

1) What is the purpose of an ASBR?

Answer: An Autonomous System Boundary Router imports routes from external networks (such as RIP routes from an attached RIP network) into OSPF. This router also exports OSPF routes into these external networks.

2) What is the purpose of an ABR?

Answer: An Area Border Router communicates OSPF routes between a site area and the backbone area (Area 0)

3) What type of OSPF network utilizes a DR?

d) BMA

Broadcast Multiple Access networks utilize a DR and BDR to conserve bandwidth and CPU cycles (updates from all other routers only need to be sent to these routers, not everyone).

4) What address does the RID default to?

Answer: The highest IP address of all interfaces configured to run OSPF. If none are configured, the highest loopback IP address is taken from those loopback addresses configured to run OSPF.

5) What type of router sends Type 2 Network LSAs on a broadcast segment?

c) DR

The Designated Router on a BMA segment is responsible for summarizing network routes and sending out Type 2 summary advertisements. In the event that the DR fails, the BDR assumes this role.

6) What type of router cannot exist in a stub area?

a) ASBR

An ASBR is intended to import and export external routes to/from the OSPF domain. These types of routes will not enter or propagate in a stub network.

7) Give 2 purposes of the Hello message.

b) Discover neighbors on a network segment.

c) Maintain neighbor adjacencies (reset timers before the dead timers expire)

8) Which State is a router in after a Hello is sent but before one is received?

   a) DOWN. Only after a Hello is received with the router's own address will that router move to INIT state.

9) What are the default OSPF timer values?

   Answer: 10 second hello timer and 40 second dead timer.

10) Which OSPF router distributes area routes into other areas?

    b) ABR

    An area border router distributes routes into other areas (most often the backbone area, Area 0)

11) Which CLI command will show the state of OSPF to other routers?

    c) show ospf neighbor

12) Which CLI command with show the type of networks the router participates in (Point to Point, BMA, etc)?

    a) show ospf interface

13) What is the CLI command to view the OSPF link-state database?

    Answer: Show ospf database

14) On an Ethernet segment Router A (priority 100), Router B and Router C (both priority 90) come online. Which one is elected to be the designated router?

    Answer: Router A has the highest priority, and therefore will become the DR.

15) On the same segment above in 14, Router D comes online 30 minutes later with a priority of 110. Assuming nothing else has changed, which router is now DR?

    Answer: Router A remains the DR as an election is not forced if a DR already exists on a segment.

16) What is the minimum configuration necessary to run OSPF on a router?

    - Which interfaces will participate in OSPF
    - Which areas those interfaces will be assigned to

17) What is the first State when establishing an OSPF adjacency?

    a) Down

    Routers start in DOWN. They move to INIT after receiving a Hello message from a neighbor with their own IP address included.

18) In which State should routers be with their neighbors on a BMA network where neither are DR or BDR?

c) 2way (indicating an adjacency, but no routes exchanged. Routes are only exchanged with the DR and BDR.)

19) What State should synchronized neighbors on a Point to Point segment be in?

f) Full

Chapter 6 IS-IS

1) The first byte of the Area ID is also known as:

d) AFI The Address Family Identifier

2) When a Level1 router encounters packets destined for a different Area, what does it do?

b) Sends it to the nearest L1/L2 router

3) What is the command to view which interfaces are configured for IS-IS?

a) show isis interface

4) By default, an IS-IS enabled interface is placed into which Level?

d) Both Level 1 and Level 2

5) What is the command to view the IS-IS database?

Show isis database

6) When configuring a router for IS-IS, the last byte of the NSAP must be set to what?

Answer: The last byte of the NSAP is known as the NSEL. The NSEL of a router must be set to 00, which denotes an Intermediate system capable of routing packets.

7) Three IS-IS routers (A, B, C) are brought online in an Ethernet segment. A has no priority configuration (default). B has priority set to 0. C has a priority set to 100.
Who will be elected the DIS?

c) Router C

Default ISIS priority within JUNOS is 64. The router with the highest priority is elected the DIS.

8) Which of the following is/are functions of a PSNP (choose all that apply): All of these.

   a) Request a missing LSP
   b) Share routing table information
   c) Notify to retransmit an LSP after a sequence error is noted
   d) Reset the Hello timer

9) Which is better thought of as a "site" router (meaning no direct communication to the network backbone)?

   a) Level 1 router

10) Identify the portions of the NSAP address 49.0001.9aff.00ab.0030.1223.4045.00

   49 = The AFI portion of the Area ID
   0001.9aff.00ab = Variable length Domain Identifier of the Area ID
   0030.1223.4045 = System ID of IP address 3.12.234.45
   00 = The router NSEL

11) Which PDU is used to advertise a complete listing of all IS-IS LSPs in the router link state database?

   a) CSNP or Complete Sequence Number PDU

12) If an IS-IS router detects a fault with its link state database, which PDU does it use to request an update?

   b) PSNP or Partial Sequence Number PDU

Chapter 7 BGP

1) At what level of the configuration hierarchy is the AS number set?

   c) [edit routing-options]

2) What do the numbers in the final column of the below output mean for the bgp neighbor?

```
10.244.2.154 65333 452 432 0 4 5d19h 8250/121636/0
```

   d) Routes active/received/dampened

3) How do you get BGP to advertise all OSPF routes to all neighbors?

   d) Import policy in BGP that matches on OSPF routes

4) Given equal cost BGP routes in inet.0 and inet.3, which will be selected?
   b) inet.0

5) Given the list of criteria for BGP route selection within JUNOS, put them in order.
   Local pref
   AS-path length
   Origin code
   MED
   eBGP vs. iBGP
   IGP cost

6) What BGP message contains information on new and changed routes?
   a) update

7) Which port does BGP listen to?
   b) TCP 179

8) What is local preference used for?
   Answer: When numerous BGP routes to a common destination are received, BGP uses criteria to select which path is best. One of the most important values is local preference, a configurable value.

9) By default, BGP will advertise what routes to neighbors (select all that apply?
   a) directly connected
   d) routes learned via BGP

10) Which command is used to view specific BGP neighbor information?
   b) show bgp neighbor

11) iBGP peers are those that are:
   a) In the same AS

12) Unless BGP multi-hop is enabled, external BGP neighbors must be:
   c) Directly connected

13) If two or more valid paths are available, BGP will inherently:
   d) Not load share.
   BGP does not load balance by default. BGP multi-path and multi-hop must be used in conjunction to facilitate load balancing.

14) iBGP neighbors must be directly connected.
    b) False.
    Internal peers must only be reachable via the IGP to be valid.

15) A router configured with BGP can automatically discover BGP neighbors.
    b) False – BGP neighbors must be explicitly configured. There is no auto-discovery feature for BGP.

16) BGP split horizon dictates that:
    d) Routes learned via iBGP are not advertised to any iBGP neighbors.
    Note that routes learned via iBGP will be passed to external peers.

17) Which two of the below help reduce the problems of requiring a full mesh of iBGP neighbors?
    a) Confederations
    d)Route reflectors

18) How does BGP avoid routing loops? (choose two)
    b) BGP split horizon ensures routes do not propagate past immediate neighbors.
    c) eBGP checks the as-path attribute and drops prefixes that include its native AS.

19) What can a MED from AS 65000 communicate to a BGP peer in AS 64666?
    a) Which interface or session would be preferred for AS 65000 to receive traffic.
    A lower number indicates a preferred path. MEDs have no explicit indication of number or location of exit points, only a relative preference. They cannot force a peer's routing decisions, only make suggestions.

<u>Chapter 8 MPLS</u>

1) What is an LSP?
    a) A unidirectional path through an internal network.

2) Which of the below are valid label operations?
    a) Pop
    c) Swap
    d) Push

3) What object specifies the next-hop of an LSP?
   b) ERO

4) What are the advantages of signaled versus static LSPs?
   a) Dynamic LSP creation
   b) Less administrative overhead

5) What functions does a router perform when it receives an MPLS packet with a single of label 0?
   d) Pop the label and forward to the IP destination .
   This special label indicates to the router to remove the label and forward based upon IPv4 address information.

6) MPLS does not require an underlying routing protocol to function.
   b) False.
   MPLS still requires an underlying protocol such as OSPF or IS-IS to allow forwarding paths for label distribution.

7) Is it possible to use an LSP as a BGP next hop?
   a) True

8) Name two MPLS protocols for signaling LSPs.
   a) RSVP
   c) LDP

9) An LSP is signaled from Router A through Router B and C to Router D. Which is the penultimate hop?
   c) Router C

10) An LSP is configured with strict constraints. The ingress router sees the ERO with a strict constraint of next hop 10.1.2.4. What must be true of this address?
    c) It must be directly connected to previous LSP hop.

11) What is contained in the RRO?
    a) A list of all hops within an LSP.

12) What is the primary purpose of the RRO?
    b) To prevent loops within an LSP.

13) What are the two ways a router can allocate label space?
    a) Per-interface
    b) Global

14) What functions does a router perform when receiving an MPLS packet with top label 1?

    c) Pop the label and forward it to the RE.

    Special label values:

    - 0, IPv4 Explicit Null Label—The label must be popped upon receipt and forwarding should continue based on the IPv4 packet address. (Note that this value is legal only when it is the sole label entry, no label stacking.)
    - 1, Router Alert Label—When this is the top label, it should be popped and the packet delivered to the local software module for processing.
    - 2, IPv6 Explicit Null Label—Same as label 0, except for IPv6
    - 3, Implicit Null Label—Indicates penultimate hop router is next hop in the LSP

15) If an LSP is configured with a secondary path and 'standby' enabled, when is the secondary path signaled?

    c) At the same time the primary is signaled.

16) When an LDP router receives multiple label bindings for a single destination what metric is used to select the best label?

    a) IGP metric

17) What information is used to compile the TED? (select all that apply)

    b) OSPF type-10 LSAs.
    d) IS-IS TLV extensions.

18) What happens if an MPLS router receives an IP packet destined for a network for which the router has no label, but has a standard IP route?

    b) The packet is forwarded via standard IP forwarding.

19) What information is stored in the inet.3 routing table?

    c) LSPs

Chapter 9 Multicast

1) What is the range of available IP Multicast addresses?
   c) 224.0.0.0 – 239.255.255.255

2) What is the purpose of IGMP?
   b) To control the hosts that join and leave multicast streams.

3) What is an advantage of IGMPv2 over IGMPv1?
   c) Version 2 uses explicit Leave messages to reduce unneeded traffic on the LAN.

4) What is the biggest advantage of Shortest Path Trees (SPTs) compared to Shared Trees (STs)?
   c) SPTs minimize latency by finding the optimal path between each source and each receiver.

5) What is an advantage of using Shared Trees?
   d) Shared Trees require less state handling and memory overhead on multicast routers.

6) What information does the router use to do an RPF check?
   a) The unicast routing table

7) Why is Protocol Independent Multicast (PIM) called Independent?
   d) It works with any IP unicast routing protocol.
   PIM works with any underlying IP unicast routing protocol—RIP, EIGRP, OSPF, BGP, or static routes.

8) What is the main advantage of MBGP?
   d) It allows for dissimilar unicast and multicast routing topologies.

9) What is the purpose of an (S, G) notation?
   a) To identify the source's unicast address with a group multicast address.

10) What two things do dense mode multicast protocols assume?
    a) That many hosts require the multicast stream.
    c) That bandwidth is plentiful.

11) What is the purpose of MSDP?

a) To exchange (S, G) notation with RPs in other autonomous systems.

12) What is the purpose of MBGP?

b) To exchange RPF information with another AS.

13) Which of the below things do sparse mode multicast protocols assume? (choose two)

b) That the RP will begin to administer the source stream to interested groups.

c) That bandwidth is scarce.

Chapter 10 Policy

1) How many firewall filters can be placed on an interface?

c) one per logical in/out

2) Select two policy actions from the below:

a) Reject

d) Accept

Push, Pop, and Swap are label actions, not policy actions.

3) If three match conditions are specified within a single accept term, when will a route be accepted?

c) When it matches all three conditions.

If multiple conditions are specified in a single term, a route must match them all in order to be considered a 'match'.

4) What is the difference between a prefix-list and a route-filter? (choose all that apply)

a) A route filter can specify an immediate action upon matching

d) A route filter can specify address ranges

Route-filters allow for more flexible address matches and can also perform an action immediately (actions can only be applied to **all** routes that match a prefix-list).

5) Routes learned from an eBGP neighbor are subject to what type of policy before entering the routing table?

d) import

6) Policy statements can be applied to which of the following?
   b) Routing Protocols

7) If no accept or reject is explicitly configured for a BGP policy term what is the default behavior for importing routes?
   Answer: Accept all

8) If a policy statement is applied to a specific BGP peer and a different policy is applied to the group to which that peer belongs, which takes effect?
   b) Peer

9) Within a policy the 'from' statements define which of the following?
   a) match conditions

10) Which of the following policy-statement operators are used to change the attributes of a route?
   b) then

11) If a route doesn't match a given term, the default behavior is to:
   a) Evaluate the route against the next term

12) Which of the following are valid match criteria within a policy-statement? (choose all that apply)
All are valid criteria
   a) source address
   b) protocol
   c) as-path
   d) next-hop

13) "192.168.64.0/19 orlonger" matches:
   b)192.168.64.0/19
   c)192.168.64.0/20

14) "192.168.64.0/19 upto /24" matches:
   b) 192.168.64.0/20
   c) 192.168.64.0/23
   d) 192.168.64.0/24

Index

About the Author

John Jacobs (JNCIA, CCNA, MCSE) started in the networking industry in the mid-1990s building local area networks. His interest soon shifted to wide-area networking and the unbounded potential held within. After working in operations for a startup DSL carrier, Rhythms Netconnections, he is currently employed as a Network Operations Engineer for a Tier 1 ISP.

Jeff Ringwelski (JNCIA, CCNA, CCNP) started networking in the late 1990s. His experience spans from maintaining small business connectivity to working in national carrier operations. He is currently a Technical Lead with a Tier 1 telecommunications service provider.

Tyler Wessels (JNCIA, CCNA) started in the networking industry in the mid-1990s working as a network technician for PSINet Inc. With a desire for a greater challenge he moved on to work for another global Tier 1 ISP as a Network Operations Engineer where he has applied and expanded his IP routing and switching knowledge.