NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCE
DEPARTMENT OF INFORMATION AND TELECOMMUNICATIONS

BSc THESIS

«Routing Protocols in Modern IP Networks»

Chrysovalantis A. Oikonomopoulos

Supervisor: **Lazaros Merakos,** Professor

ATHENS

OCTOBER 2019

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«Πρωτόκολλα Δρομολόγησης σε Σύγχρονα IP Δίκτυα»

Χρυσοβαλάντης Α. Οικονομόπουλος

Επιβλέπων: **Λάζαρος Μεράκος,** Καθηγητής

ΑΘΗΝΑ

ΟΚΤΩΒΡΙΟΣ 2019

BSc THESIS

«Routing Protocols in Modern IP Networks»

Chrysovalantis A. Oikonomopoulos
S.N.: 1115200600086

SUPERVISOR: **Lazaros Merakos, Professor**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

« Πρωτόκολλα Δρομολόγησης σε Σύγχρονα IP Δίκτυα »

Χρυσοβαλάντης Α. Οικονομόπουλος

Α.Μ.: 1115200600086

ΕΠΙΒΛΕΠΟΝΤΕΣ: **ΛΑΖΑΡΟΣ ΜΕΡΑΚΟΣ,** Καθηγητής

ABSTRACT

Modern IP networks are continuously evolving and growing. The fact that more and more devices become “smart” and have the ability to connect to an IP network makes network engineers come across a variety of different network topologies, on a daily basis, interconnecting hundreds or thousands of different subnets. IP routing is the key link between these subnets. The purpose of this thesis is to become a reference tool for students or engineers whose main responsibility is the management or administration of core routing technologies.

SUBJECT AREA: Routing Protocols

KEYWORDS: OSI Model, IP, Routing, IPv4, IPv6, OSPF, RIP, EIGRP, BGP

ΠΕΡΙΛΗΨΗ

Τα σύγχρονα IP δίκτυα συνεχώς εξελίσσονται και μεγαλώνουν. Ο αυξανόμενος αριθμός των όλο και περισσότερο ο διασυνδεδεμένων "έξυπνων" συσκευών, υποχρεώνει τους μηχανικούς δικτύων να πρέπει να διαχειριστούν ποικίλα δίκτυα με εκατοντάδες ή χιλιάδες διασυνδεμένες συσκευές. Η δρομολόγηση του IP πρωτοκόλλου είναι ο συνδετικός κρίκος μεταξύ όλων αυτών των δικτύων. Σκοπός της παρούσας πτυχιακής εργασίας είναι να αποτελέσει ένα εργαλείο αναφοράς των πρωτόκολλων δρομολόγησης, για σπουδαστές και μηχανικούς, των οποίων κύρια δραστηριότητα είναι η διαχείριση και η εποπτεία τεχνολογιών και πρωτοκόλλων δρομολόγησης σε IP δίκτυα.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Πρωτόκολλα Δρομολόγησης

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Μοντέλο OSI, RIP, OSPF, EIGRP, ISIS, BGP

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF IMAGES

LIST OF TABLES

PREFACE

My thesis “Routing Protocols in Modern IP Networks” was an inspiration during my first years in university. From my early academic years, I was working as a network engineer and instructor too. My intend for the current thesis was to create a quick reference guide for routing protocols for both students and engineers. IP Routing is a vital service for any IP network in our world. The more we know about routing protocols – the better we can handle the routing and forwarding decisions in our networks.

During the writing process, I used a variety of tools in order to have better graphic representation of topologies and I tried to keep away technical configuration because it is vendor specific at the most. However, when it was necessary to add examples of configuration that was from Cisco Router IOS version 12.4. Additional diagrams or network topologies have been created using either Cisco Packet Tracer or the GNS3 network emulators.

At this point, i would like to give special thanks to my professor, Mr. Lazaros Merakos for supporting me all these academic years with his valuable guidance through both scientific and professional advice. Except of his expertise in networking technologies, his great personality also gave me a different point of view both in university and in networking field.

Last but not least, I would like to thank all the Scientific and Faculty’s for the excellent cooperation all these years.

Lastly, the biggest thank you is to my family for the undivided love, dedication and support they provide me at every step of my life.

1. INTRODUCTION

1.1 Living in a Network-Centric World

Throughout history, human civilizations have depended on the structure of various community networks for safety, food, trade and companionship. At one time, sounds, smoke signals and gestures were all humans used to communicate. Later on, people used paper and drawings to exchange information and the post service to transmit it. Now, the technical developments with the Internet allows people to instantaneously share all types of communication, for example documents, pictures, voice and video with thousands of people near or miles away using computers.

The evolution of Information Technology is, perhaps, the most significant change agent in the world today, as it helps to create a world in which national borders, geographic distances and physical limitations become less relevant and present ever-diminishing obstacles. Networks and networking have grown exponentially over the last 15 years. They have had to evolve at light speed just to keep up with the huge increases in basic mission critical user needs such as sharing data and network devices, as well as more advanced demands such as voice over ip, videoconferencing or telepresence.

1.2 IoE: Internet of Everything

Few years ago, only computers were able to connect to a network but, today, almost any electrical device can be connected to a network. Smart-phones, tablets, cars, security cameras, smart-tv, smart-watches, door-locks, central heating-cooling systems, interior lighting and most home appliances can, now, be controlled through one device and connected to a private or public network infrastructure using special software. In the near future, Internet of Everything will be a reality. People will control their homes, their cars even their hole factory production remotely from everywhere on the earth. The fact that, year by year, even more devices can be connected to a network and share their data with it make us to ask ourselves: "*how this enormous amount of data is transmitted and controlled in the mess of the worldwide network and ever more, how secure is all this information?"* The key to the answer is that data networks are structured.

In the next pages, first, we are going to analyze this structure and, then present the technologies used to transport data through various devices throughout the world.

2. THE NETWORK ARCHITECTURE

2.1 The Role of the Network in IT

An Information Technology infrastructure system consists of a number of different devices and technologies and is designed to offer some services to its administrator. If we try to analyze IT infrastructure system, we will found out that it consists of many sub-systems, each one having its own role and providing certain services. Its sub-system requires different knowledge and management and most of the times it is built by different people. Almost every IT infrastructure contains three elements:

- **Software:** The software is a set of instructions that enable a user to interact with a device, mostly a computer, and assign to it specific tasks to implement. Software is build by programmers.
- **Systems/Servers:** System is a collection of different software united in a single entity. Its purpose is to offer a variety of services. Usually, we use the term operating system and we have different operating systems based on their owners and their purpose. An operating system is installed on a computer. If this computer is intended for personal use, then we install a basic operating system like Linux (open source) or Windows (Microsoft) or any other version. If the computer is intended for professional use, then we have to use the server edition of the operating system which make a much more efficient use of resources and can handle a huge amount of data. Most common use of servers is to keep user profiles, maintain databases, make complicated calculations, run programs, host webpages or files, etc. Systems are managed by system administrators and programmers.
- **Network:** Any IT system has one goal: to produce data. These data may stay local on the machine but most of the times must travel to another device. For example, we can have database queries and exports, email delivery, chats, video conference, upload or download files from FTP servers etc. Network's role is to provide all means for the data to travel from one device's network interface (NIC) card to another device's NIC. The network defines all the necessary physical components and software in order to deliver data from one point to another. In other words, the network is the communication channel between the different devices. Either the devices are directly connected or miles away network's job is to provide reliable, secure and fast delivery of the data between them. We all understand that a total network failure results in total isolation of our hole infrastructure. A network is managed by a network administrator.

// At this point, we must admit that from network administrator's perspective, we "don't care" about the data we receive from the software. We "can't see" if there is a "good" program or a virus. The only thing that concerns us, is "what type of data" we have to transmit. //

2.2 Elements of a Network

The model of sending a message through a channel to the receiver is the basis of network communication between computers. The computers encode the message into binary signals and transport them across a cable or through a wireless media to the receiver, which is governed by the same rules with the source in order to understand and decode the received message. Any type of network has the above common elements [1]:

Message source -> Encoder -> Transmitter -> Channel -> Receiver -> Decoder -> ->Message destination

The message source or destination, including the encoder-decoder and the transmitter-receiver, is a piece of equipment to which we usually refer as *host* or *end device*. Examples of end devices are computers, servers, network printers, voip phones, network cameras, pdas, laptops and their users. Most of the times we don't care about the type of the end device but for the number of the end devices we are going to have in our network in order to design a correct ip addressing scheme. Except of the end devices, in a network we will, also, find *intermediary devices*. Intermediary devices are all the devices that connect the hosts to the network or connect different networks to form the Internetwork. Example of intermediary devices are routers, switches, hubs, modems, wireless access points etc.

The channel is the medium over which the data travel from the source to the destination. The three main types of media in use in a network are:

- Copper cable
- Fiber-optic cable
- Wireless

Each of these media has different physical properties and requirements and uses different methods to encode messages. Encoding and decoding refers to the way the data is converted to patterns of electrical pulses, light pulses, or electromagnetic waves and carried over the medium to the destination.

2.3 Different Types of Networks

Networks can be categorized to many groups but, mainly, we categorize them due to their size of area covered, function or number of users. In daily life, we refer to 4 types of networks based on their size of area covered. These types are:

- **Local – Area Network (LAN):** A LAN is a group of interconnected devices under common administration (e.x the private network of a company or an organization), located on one floor or building.
- **Wireless LAN (WLAN):** Same type as LAN but in this case the devices are connected through a wireless network.
- **Wide – Area Network (WAN):** A WAN refers to a group of interconnected LANs located geographically far apart (e.x a company with many branches to different towns in a country or around the world). Most often the lines between the LANs are rented from a telecommunications service provider (TSP) which is responsible for the availability of these leased lines. The LANs, at each end, remain to the control of the company. Sometimes a large WAN is called intranet because it is like a private web of networks closed to the public but open for the company employees to browse.

- **Internet:** The term Internet refers to the huge network that connects all these millions of private WANs and LANs with each other. Access to the Internet is given to customers by an Internet Service Providers (ISP). Every ISP is connected to all other TSPs or ISPs worldwide in order to synthesize the worldwide Internet.

2.4 Protocols vs Standards

Every network, regardless of its size, is a community of devices where each of them wants to communicate with another one. Any community and any kind of communication on earth, in order to be successful, is governed by “rules” and data networks will not be an exemption. In our case “rules” are divided into 2 categories: a) *Protocols* and b) *Standards*. Most people confuse the two definitions and think of them as the same think, something that is completely wrong. A Protocol is set of rules created for specific equipment and it can be used only by its creator. For example, company A invents protocol ABC which is a new way of cable communication of speeds of terabytes. That means that only company A has the rights to use it and only its equipment will be compatible with it. This is called a proprietary protocol. But, if company A decides to offer the protocol ABC for free to others, then the protocol will be a “candidate” to be a standard. The organizations that standardize networking protocols are the *Institute of Electrical and Electronics Engineers (IEEE)* and the *Internet Engineering Task Force (IETF)*. After becoming a Standard, then it is widely available for everyone to include to its equipment. Standards enforce interoperability between vendors.

2.5 Using Layered Models

To successfully describe the complex process of networking communications, the IT industry invented the *layered models*. The layered concept of networking was, at first, developed to accommodate changes in technology. Network models define a set of manageable layers and how they interact. Each layer is responsible for a different function on a network and independent from the other layers. For example, a new operating system doesn’t affect the ip addressing scheme or the upgrade from Ethernet to Fast-ethernet didn’t affect the TCP/UDP protocols. All layers must handle and pass information up and down to next subsequent layer as data is processed. Understanding the layered models is the first thing to do when you are entering the IT world. This knowledge offers you a great “image” of what is happening on the background of your device.

Nowadays, networking professionals use two networking models, one protocol model and one reference model. Both of them were created in the 1970s. The first is the TCP/IP model which describes the functions that occur at each layer of protocols within the TCP/IP suite. The second one, the reference model, offers a clearly understanding of the functions and processes involved within all types of network protocols and services. The Open Systems Interconnection (OSI) model is the most widely known internetwork reference model. The OSI model describes the entire communication process in detail and the TCP/IP model describes the communication process in terms of the TCP/IP protocol suite. In the above table we can see the two models with their layers:

Table 1: OSI and TCP/IP Model

	OSI Model	***TCP/IP Model***
7.	Application	Application
6.	Presentation	
5.	Session	
4.	Transport	Transport
3.	Network	Internet
2.	Data Link	Network Access
1.	Physical	

In the next paragraphs we are going to analyze the OSI model which refers to the function of each layer and not to protocols function, so we are going to leave aside the TCP/IP suite for now. Useful information about the TCP/IP suite can be found in Request for Comments *(RFC) 1180*.

2.5.1 The Open Systems Interconnection (OSI) Model

The OSI model provides an abstract description of the network communication process. It is developed by the *International Organization for Standardization (ISO)* to provide a roadmap for protocol development. Each layer has a certain function and it provides data services to the layer above by preparing information coming down the model or going up.

Let's take a brief look at each layer, from top to down, and its goal [1]:

- **Application:** The application layer provides users an interface in order to use the network device. It includes high-level end user software which uses common application layer protocols to achieve the desired communication, for example, web browsers use http and https to get a web page. Some of the most common application layer protocols are http, dns, pop, smtp, ftp, dhcp, ssh.
- **Presentation:** This layer provides formatting functions for application layer including character coding and conversion, data compression and encryption / decryption. Some of the well-known formats are mpeg. Jpeg, mp4, giff, flash player.
- **Session:** Establishes and controls communication between source and destination applications. The session layer handles the exchange of information to initiate dialogs, keep them active and recover them in case of disruption or idle time.
- **Transport:** The transport layer provides transparent transfer of data between end-user applications providing either reliable or best-effort delivery of packets.

Firstly, transport layer receives the data for transmission from upper layers. Then, segments data into pieces called segments or datagrams. Each of them is then treated individually. Next, the transport layer must identify whether the segments are for reliable delivery or they are for fast delivery. Different types of data need different handling inside a network. Basically, we can identify 3 types of data: a) *user data* such as emails, pdfs, files etc require reliable– guaranteed delivery b) *video/voice traffic* such as voip packets, video calling, live streaming etc require very fast delivery and c) *management traffic* which is traffic exchanged between applications, protocols and devices such as authentication or keep-alive mechanisms requires, most of times reliable delivery. Transport layer has 2 protocols to offer the above services: *Transmission Control Protocol (TCP)* and *User Datagram Protocol (UDP).* TCP offers reliable delivery, flow control, error recovery and same order delivery mechanisms. Sadly, it adds a header of 20 bytes into each segment, something that increases overhead and slows down the transmission. At the other side, UDP offers fast delivery of packets but has none of the above mechanisms and it adds only an 8-byte header on the data. Data transmitted with TCP are called segments and with UDP are called datagrams. Either TCP or UDP, transport layer has one more important mission. Transport layer is responsible to identify the source and destination applications that communicate. To accomplish this, it assigns a unique identifier to each application, called *port number*. Any application running in a device is assigned a unique number between 1 – 65535 when it is initiated. Some numbers are reserved for well-known protocols or applications. The different types of port numbers are:

- Well-known ports: 0 – 1023
- Registered ports: 1024 – 49151
- Dynamic or private ports: 49152 – 65535

Well-known and registered ports have been assigned to known protocols and applications by *Internet Assigned Numbers Authority (IANA)*. Dynamic ports are reserved from the operating system when an application starts, for example every time we open a new tab in the web browser, the operating system reserves a new port. Transport layer writes in each packet’s header the source and the destination application port numbers.

- **Network:** The network layer’s main purpose is to provide all the services needed to forward a packet from one device to another. To accomplish this mission, network layer offers two basic services: *addressing* and *routing*. For two devices to communicate there must be a kind of addressing, so packet delivery is based on a source and destination address. The type of addressing depends on the used protocol. Known network layer protocols are IP (version 4 & 6), AppleTalk, CLNS and IPX. Internet Protocol is the most widely used Layer 3 data-carrying protocol. It offers each device a unique address, inside a network, either IPv4 or IPv6 address, which also called logical address because it can be different from one network to another. The last decades, IPv4 (32-bit addressing) has been used, almost, in every network and today it gives its place to the new IPv6 (128-bit addressing). IP is a connectionless, best-effort protocol which runs over any type of network and encapsulates TCP or UDP segments with all the necessary information in order to travel to the destination device. The way a packet is going to reach its destination is also network layer’s responsibility and is called routing. A packet travels through various networks based on its source and destination IP address. The devices responsible for the forwarding decisions are called *Routers*. Routers main job is to determine a network and to find out the best routes for a packet to reach other networks. The process of learning new networks and calculate the best routes to them is called *routing*. We have two types of routing: *static* and *dynamic*. In static routing administrator must statically determine the paths in a network while in dynamic routing the "heavy work" is done by dynamic routing protocols like Rip, Eigrp, Ospf etc.

- **Data Link:** The data link layer defines the protocols required to deliver data across a physical network. It links the upper-layer services responsible for packaging the data for communication between devices with the services to transfer that data across the media. Due to the variety of physical media, a wide variety of layer 2 protocols define different types of frames and different methods of controlling access to the media. Data link layer performs two basic functions: a) allows the uppers layers to access each media using techniques such as framing and b) controls how data is placed into media and received from the media using techniques such as framing and error detection. To support these functions, data link is divided into two main sub-layers: a) *Logical Link Control (LLC)* which places information in the frame that identifies which network layer is used, and b) *Media Access Control (MAC)* which provides layer 2 addressing (also called physical addressing) and delimiting of data according to the physical signaling requirements of the medium and the type of data link layer protocol in use. In order to achieve frame delimiting, except from header, layer 2 protocols add a trailer at the end of the frame. The trailer contains a field which define the end of the frame and a *Frame Check Sequence (FCS)* field which is responsible to detect errors in the frame. Common data link layer protocols are 802.3 Ethernet, PPP, ATM, HDLC, ARP, 802.11 a/b/g/n/ac, LLDP, STP, Frame - Relay etc.

- **Physical:** Physical layer provides the means to transport across a network media the bits that make up a data link layer frame. The main functions of the physical layer include: a) *the physical components* – the hardware devices, media and connectors that transmit and carry the signals to represent the bits, b) Data encoding – a method of converting a stream of data bits into a predefined "code". Codes are grouping of bits used to provide a predictable pattern that can be recognized by both the sender and the receiver, and c) Signaling – physical layer must generate the electrical, optical or wireless signals that represent the "1" and "0" on the media.

The understanding of the layered approach, either OSI or TCP/IP model, is a very good start, in order to understand how a network is working. Also, is a very good way to troubleshoot common, daily, problems in a network. Starting from down to top and checking each layer's technologies and protocols, especially in OSI model, is the best way to catch out and isolate the problem in an end to end communication.

2.5.2 Follow that frame.... Encapsulation – Decapsulation Process

Based on the analysis of the layered approach models, we are going to analyze what exactly happens and how a frame is delivered, from the networking aspect, when two or more devices are trying to communicate with each other.

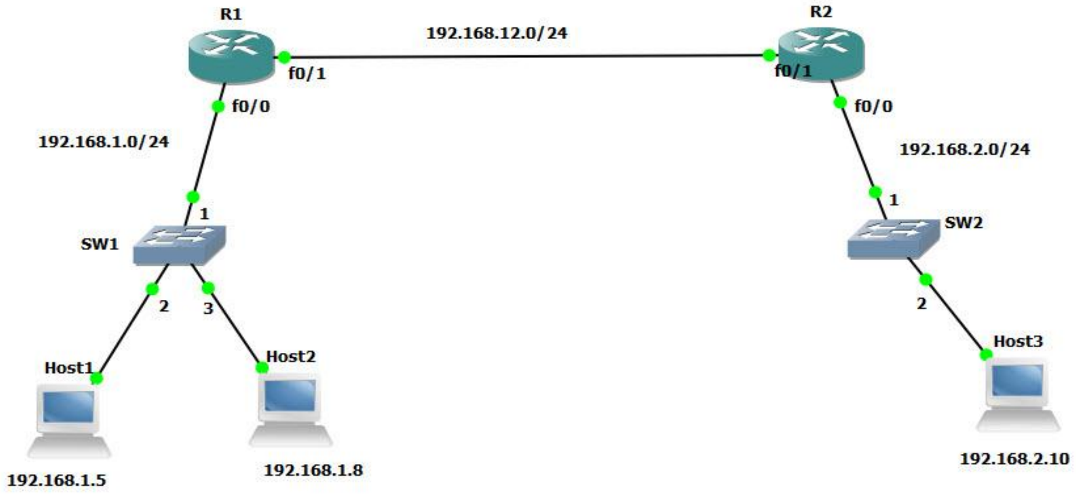


Figure 1: Topology Example 1

In the above network topology [Figure 1], we see three networks – subnets, one for router to router connectivity and two for users. We assume that all devices are fully configured, the links between devices are all UTP connections and each host uses a messaging application to communicate with other users. We will analyze two scenarios 1) Host1 to Host2 communication, and 2) Host1 to Host3 communication.

<u>1st Scenario:</u>

1) Host1 is writing a message to host2. After the user presses “enter” to its pc, the software of the network interface card (NIC) is receiving the message – data for transmission. We pass by the upper three layers of the OSI model and we start the analysis of the Transport layer. The Transport layer will see that it must use TCP protocol, as we have a text for transmission and not voice/video, and it will segment the data into pieces, such big as the MTU value has determined from the Network layer. Each individual piece now will be encapsulated with a TCP header and marked with the source and destination ports, in order to determine which applications are communicating. For our example purposes, we assume that source port is 62001 and destination port is 63001.

2) Transport layer passes each piece (called segment), to the Network layer for further handling. Network layer adds the IPv4 header to each piece and marks each one

with the source and destination ip address. At this scenario, source address will be the 192.168.1.5 and the destination the 192.168.1.8.

3) Network layer passes each piece (called packet), to the Data link layer. At first, data link layer will check the exit interface's protocol, utp connection uses Ethernet protocol, so will add at each packet the Ethernet's header and trailer. Ethernet requires source and destination mac-address to be written in the header. Data link layer knows the source mac-address but it doesn't know the destination mac-address. At this point, the encapsulation process of the packet stops for a while. The NIC must learn Host2's mac-address. For that purpose, the NIC will use the Address Resolution Protocol (ARP). Host1 will broadcast** an ARP-request asking for the mac-address of the user with the ip 192.168.1.8. Host2 will receive this request and it must send back an ARP-reply with its mac-address.

** In modern networks there are four types of communication:

- ***Unicast:*** one to one
- **Multicast:** one to many (a group of users)
- **Anycast:** one to nearest (used in IPv6, the same ip is configured to many devices)
- **Broadcast:** one to all (inside a network)

The actual frame leaving host1 will have the below structure:

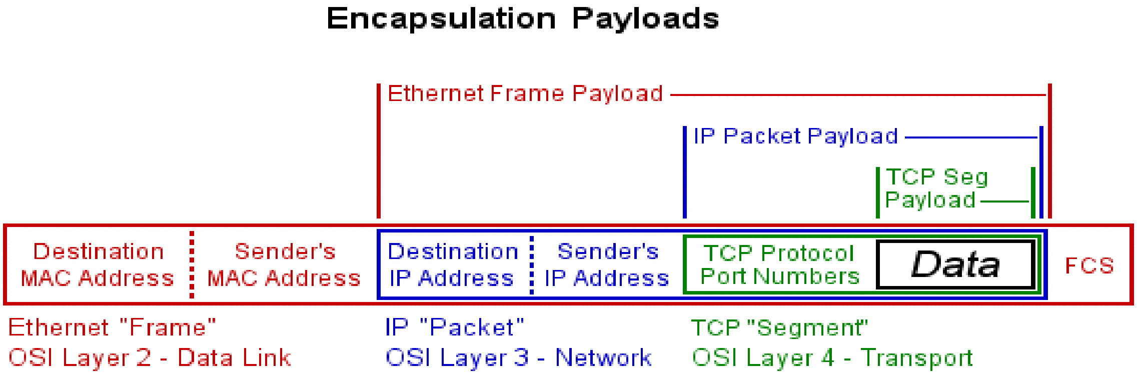
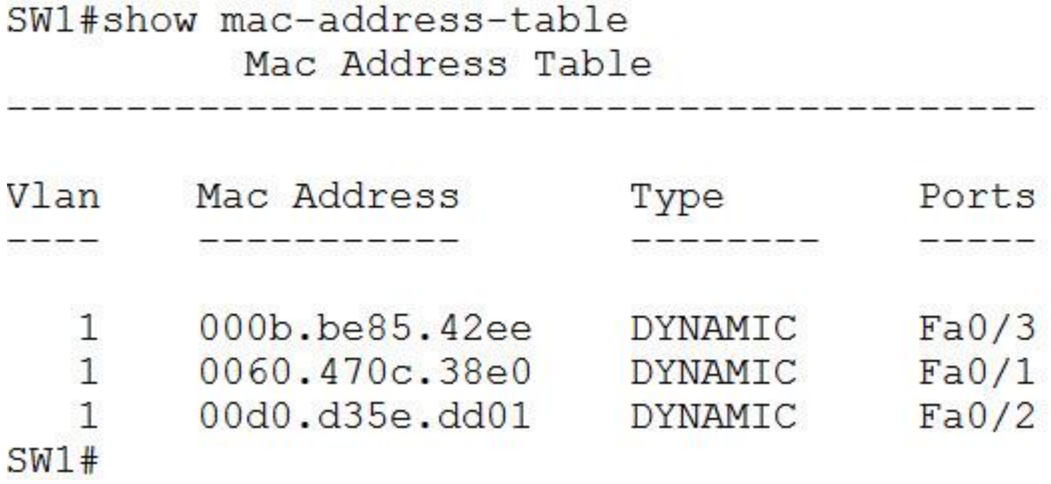


Figure 2: Structure of a frame

4) Host1 will learn the destination mac-address of Host2 and will forward the frame out of its NIC. Before sending the frame out, Host1 will run the *Cyclic Redundancy Check* (CRC) algorithm to the whole frame and the result will be written in the *Frame Check Sequence* (FCS) field, at the Trailer of the frame. Next stop is the switch (SW1). A switch mainly "works" at layer 2 of the OSI model, so it can only "understand" layer 2 protocols (L2 switch). Some switches can, also, "read" Network layer's header and can offer routing and other L3 services (L3 switch). Here, the L2 switch will forward the frame based on the source and destination mac-addresses written in the frame. To achieve this, it holds a "mac-address table" where it keeps records about mac-addresses and the outgoing interfaces. An example of a mac-address table we see on Figure 3:



```
SW1#show mac-address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----

   1    000b.be85.42ee    DYNAMIC     Fa0/3
   1    0060.470c.38e0    DYNAMIC     Fa0/1
   1    00d0.d35e.dd01    DYNAMIC     Fa0/2
SW1#
```

Image 1: Mac-address table of a switch

Each time a frame passes through a port, the switch makes an entry in the table with the source mac-address and the interface the frame was received. Now, the SW1 will use the destination mac-address written on the frame and it will forward the frame out of the interface fastethernet 0/3 to Host2.

5) Host2 will receive the frame and it will start the decapsulation process. First of all, it will run CRC algorithm, again, in order to check for errors in the frame. The result must be the same with the value written in the FCS field of the frame, otherwise the frame will be considered to have errors and host2 will drop it. Next, host2 will check the destination mac-address field that contains its mac-address, it will delete the Ethernet header and it will pass the data to the network layer. It will check, then, that the destination ip address field has its own ip address, it will delete the ip header and it will pass the packet to the transport layer where the TCP protocol will wait to receive all the segments, in order to reassemble the whole data. Finally, by checking the destination port number, TCP will pass the data to the appropriate software.

2^nd^ Scenario:

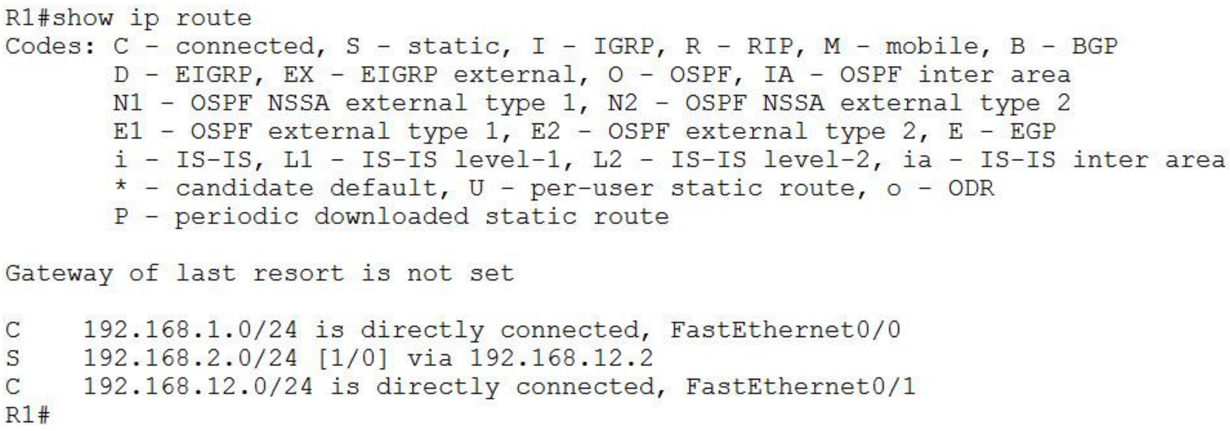
1) The first 3 steps from scenario one are repeated, now with a destination ip of 192.168.2.10, until the encapsulation process reaches the point where host1 needs to learn the destination mac-address for host3. At this point, a broadcast arp-request for the ip 192.168.2.10 would be meaningless because this broadcast would never reach host3, as it belongs to a different broadcast domain (**routers are delimiting broadcast domains - a router would never pass a broadcast packet from one network to another). So, how host1 will learn host3's mac-address? The answer is that either we have to enable Proxy ARP in R1 or we have to configure host1 with a default gateway of 192.168.1.1 in order for the encapsulation to be completed. In case of Proxy ARP feature, as R1 receives the arp-request for 192.168.2.10 it will understand that no one will answer this request because, as a router, knows that this ip belongs to another network. Also, R1 "knows" how to reach that network (192.168.2.x), so it pretends to be host3 and responds to the first arp-request with its own mac-address. However, due to the fact that leaving Proxy ARP on, in a network, is not a good practice for security reasons, the most preferred solution is to set up a default gateway to host1. For a network device, the default gateway is its way out to other networks, which is network's router. By configuring a default gateway to host1, when host1 will have a frame for another network (** the pc based on

its own ip and subnet mask, can figure out whether the destination ip is in the same or in another network) it will "see" immediately that it must send that frame to its default gateway. So, the host1 will write in the destination mac-address field of the frame, the mac-address which matches with the default gateway - in our case router's mac-address. Both options result to use as destination mac-address router's mac-address.

2) Host1 sends out the frame to SW1. SW1 will receive the frame and based on frame's destination mac-address and its mac-address table, it will forward the packet out to R1.

3) As soon as R1 receives the frame in fastethernet0/0 port, it checks the frame against three parameters:

- Destination mac-address which should be its own mac
- Calculates CRC for error checking
- If the frame passes successfully the above two checks then the router strips off the data link layer header and reads the destination ip field of the ip header. Router will check the destination ip of the packet against its routing table. Routing table is a dynamic table where routers keep info about the networks they know and how to reach them. R1’s routing table is shown below.



```
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, FastEthernet0/0
S    192.168.2.0/24 [1/0] via 192.168.12.2
C    192.168.12.0/24 is directly connected, FastEthernet0/1
R1#
```

Image 2: R1's Routing Table

Based on figure 4, R1 will forward the frame out its fast Ethernet 0/1 port, to the ip 192.168.12.2, which is the R2’s interface ip. After the routing decision is taken, the router checks the outgoing interface protocol in order to add the appropriate L2 header to the packet. It is a fast Ethernet link, so R1 will add its mac-address as the source and R2’s mac-address as the destination address and it will calculate a new CRC value for that frame.

4) R2 will receive the frame and steps 2 and 3 will be completed in reverse order until Host3 receives the frame. At Host3 the decapsulation process will be the same as in step 5 from the first scenario.

It must be mentioned that the above analysis is focused on a general description of how data are encapsulated in a frame, travel through a network and are decapsulated in the other end. Facts like, fragmentation, quality of service, how hosts are assigned ip or how the routing table is build at R1, R2 will be discussed more in the next chapters.

3. ROUTING FUNDAMENTALS

3.1 Introduction to Routing

The main job of routers, inside a network, is to route packets between different networks. Based on its routing table, a router has to find the best exit for the received packet to its destination. By default, a router has in its routing table only the networks that are directly connected to. If a network does not exist in a router's routing table, the router cannot forward traffic to it. The term of routing or *routing process* refers to how a router is going to build, maintain and update its routing table. Building and maintain a routing table for a small, flat, network is an easy process but speaking for a large network can be a very complex task.

Routers have to learn new networks, calculate the best paths to them and keep separate tables for best and backup routes for each destination. There exist two types of routing:

- ***Static Routing*****:** the administrator must configure manually the router with all the networks, inside a topology, and the preferred paths to them. Static routing is more secure, consumes no CPU or Ram but it scales well only in very small networks and requires the presence of the administrator in order for routing changes to be made.

- ***Dynamic Routing*****:** a dynamic routing protocol is a set of processes, algorithms and messages that are used to exchange routing information, choosing the best path to a destination and maintaining up to date routing information. The network administrator chooses, configures and optimizes a dynamic routing protocol that best fits the network topology and the routers automatically exchange routing information between each other. Algorithms handle the exchange of networks and choose the best path to each of them. Whenever a change happens to the network, the routing protocol will take all the necessary actions to adjust the topology to the new change and keep the network up. To achieve this level of automation, dynamic routing protocols consume a great amount of CPU and Ram from the router and require very good knowledge of them in order to configure, verify and troubleshoot them.

Dynamic routing protocols are used by routers to share information about the status and the reachability of remote networks. There are several dynamic routing protocols for IP, divided in categories based on their characteristics [6]:

- **Interior or Exterior Gateway Protocols**
- **Distance Vector or Link State**
- **Classful or Classless**

3.1.1 Interior and Exterior Gateway Protocols

An *Autonomous System (AS)* [6] is a collection of IP routing prefixes under a common administration. Typically, an AS is either an Internet Service Provider (ISP) or very large organization like a university or an international company, with independent connections to multiple networks. Each AS is allocated a unique number called, the Autonomous System number (ASN). Every entity in the Internet is known via each ASN. Older ASN system used a 16-bit integer and allowed 65.536 assignments, is replaced by the new version, defined in *RFC 4893*, which uses a 32-bit integer and allows about 4.294.967.295 ASNs. Internet is based on the autonomous system concept and that's why two types of routing protocols exist:

- *Interior Gateway Protocols (IGP):* Used for routing inside an autonomous system.
- *Exterior Gateway Protocols (EGP):* Used for routing between different autonomous systems. The only available EGP protocol is Border Gateway Protocol (BGP).

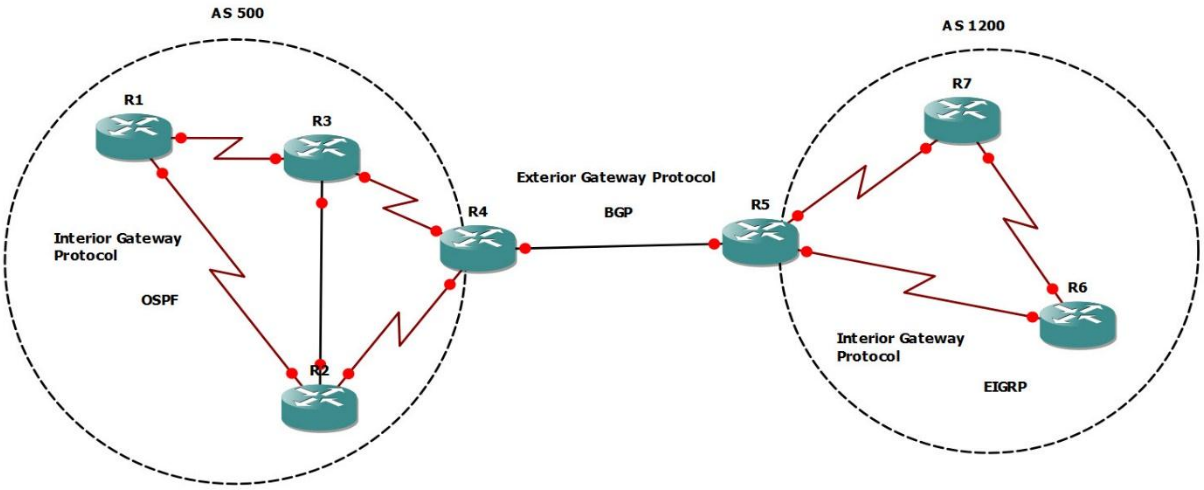

AS 500
R1
R3
R4
R2
Interior Gateway Protocol
OSPF
Exterior Gateway Protocol
BGP
AS 1200
R7
R5
R6
Interior Gateway Protocol
EIGRP

Figure 3: Routing between AS

3.1.2 Distance Vector and Link State

Interior gateway protocols can be categorized in two types [6]:

- *Distance Vector:* in a distance vector protocol routes are advertised as vectors of distance and direction. A router, running a distance vector protocol, learns about a remote network, a metric value (distance) and which path or interface should use to get there (direction). Distance vector protocols cannot build a map of the network topology. Also, routers running distance vector protocols base their routing policy on neighbor's routing policy which means that they don't know anything about the topology after their neighbor. This happens because routers running distance vector protocols send only their routing tables to their neighbors which include only their best routes. However, a distance vector protocol is easier to configure, consumes less resources from the router than a link state and does not require hierarchical design. Distance vector routing protocols are: RIP, RIPv2, IGRP(obsolete) and EIGRP.

- *Link State:* in contrast to a distance vector protocol operation, link state protocols create a complete view of the network by gathering information from all other routers. Each router populates its networks, its directly connected neighbors, its interface bandwidth and other info about itself to all routers in the routing domain. Routers in the routing domain receive the information and stored in a database. When all routers have identical databases, each router runs its protocol's algorithm over the database and calculates its best routes and a "map" of the network. Link state protocols are also called "shortest-path-first" protocols. The whole operation makes link state protocols to require a lot CPU and Ram and an administrator with good knowledge of the implementation, but they scale very well in large networks with hierarchical design and offer fast convergence of the network. Most known routing link-state protocols are IS-IS and OSPF.

3.1.3 Classful and Classless

An old but still valid categorization of routing protocols. Dynamic routing protocols in order to exchange routing information, they create updates with their routing tables and exchange them. *Classful routing* protocols do not include the subnet mask in routing updates. When a neighbor router receives an update, it assumes the default subnet mask for a network based on its class. As a result, a received network with a network address of 192.168.x.x will be seen as a /24 network and a 172.16.x.x as a /16. This routing behavior cannot work correctly in today networks due to Classless Inter Domain Routing (CIDR) which uses Variable Length Subnet Mask (VLSM) and creates different prefixes. Classful routing protocols are the old RIP and IGRP (obsolete). On the other hand, Classless routing protocols include the subnet mask in the routing updates. Classless routing protocols are the RIPv2, EIGRP, OSPF, IS-IS and BGP and are used in almost any network today.

3.1.4 Basic Routing Concepts

Before the routing process can determine which route to use when forwarding a packet, it must first determine which routes to include in the routing table. All the routing decisions, are based on two parameters [6]:

- **Administrative Distance and Metric**
- **Routing Table**

3.1.5 Administrative Distance and Metric

These two values control which routes are going to be entered in the routing table of a router. When a router has two or more paths to a destination, at first it compares their administrative distances and if it comes to a draw, it compares their metric values.

- *Administrative Distance:* Every routing source has a default administrative distance (AD) value which informs the router about its reliability. When a router receives the same routing information from multiple routing sources, the router compares the administrative distances of the routes in order to determine which route to install into the routing table. The value ranges from 0 – 255, where the lowest administrative distance indicates the most reliable source. An AD of 255 means the router will not believe the source of that route and it will not be installed in the routing table. Administrative Distance can be edit either via the protocol itself or by the use of methods of routing update manipulation. Table 6 contains the administrative distances of most protocols.

- *Metric:* when a router learns more than one path to a network, from the same routing source, it must determine which path is the shortest to the destination. Metric is the parameter which represents the “distance” to a destination network. The way that it is calculated depends on the running routing protocol. Some protocols use the hop count to calculate the distance where other protocols use bandwidth, delay, interface load, MTU, reliability or Autonomous System Path.

Table 2: Well Known Administrative Distances

Routing Protocol	**Administrative Distance**
Directly Connected Interface	0
Static Route	1
Eigrp Summary	5
External Bgp	20
Internal Eigrp	90
Ospf	110
Is-Is	115
Rip v1 & v2	120
On Demand Routing (ODR)	160
External Eigrp	170
Internal Bgp	200

Next Hop Resolution Protocol	250

In case that two or more routes have the same administrative distance and metric to a destination, the router will inject all of them in the routing table and will *load balancing* the frames between the equal-cost routes.

3.1.6 Routing Table

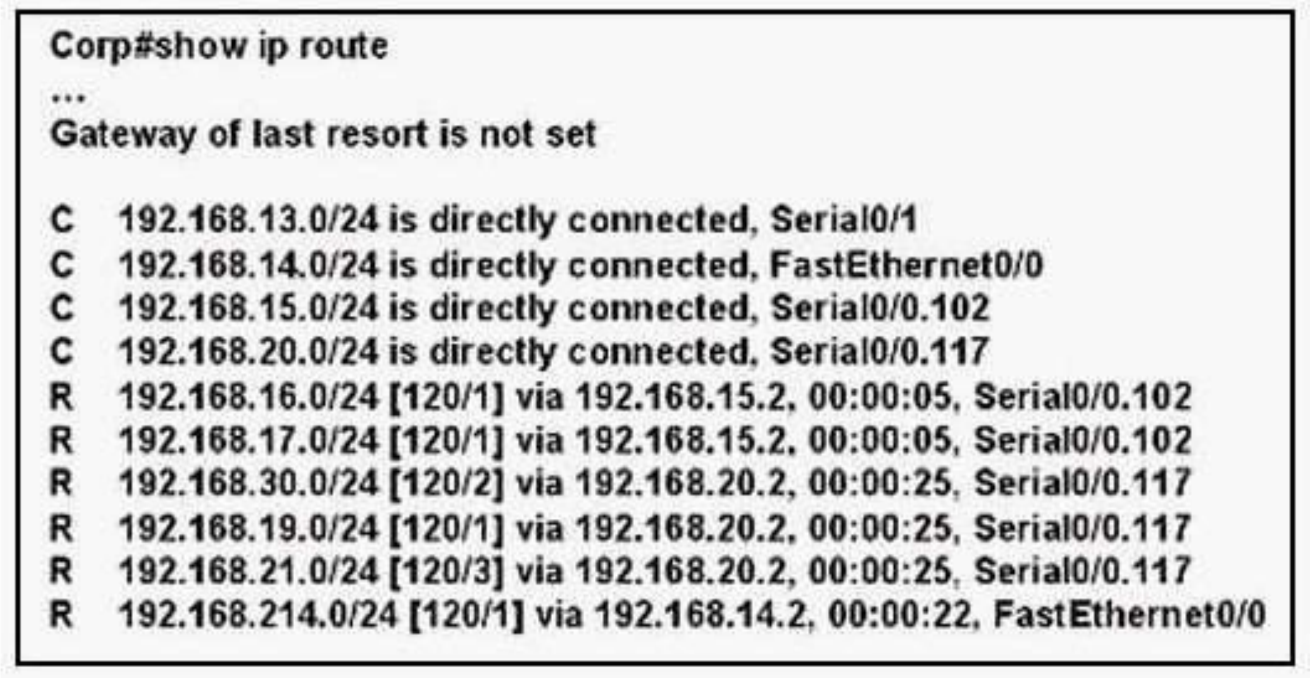
When a packet enters a router, the router strips off the layer 2 header and reads the destination ip field of the layer 3 header. Then, the router “examines” the routing table to find a matching entry for that ip address. If the router finds a valid entry in its routing table, it will encapsulate again the packet and it will be forwarded through the outgoing interface. Otherwise, it will be dropped. That makes the routing table a very important section of the routing process. One false in a routing table entry may cause a whole country to lost connectivity to a destination.

When configuring or troubleshooting a routing table we must keep in mind three basic principles of routing tables [2]:

- Every router makes its forwarding decisions alone, based on the information it has in its own routing table.
- The fact that one router has certain information in its routing table does not mean that the other routers, in the topology, have the same information.
- Routing information about a path from one network to another does not provide routing information about the return path.

Sometimes network engineers disregard the last bullet which can cause either black holes in the network, but this is a problem that it would easily noticed, or, worse, would cause asymmetric routing in a topology.

When a router boots up, its routing table contains only its directly connected networks. When routing configuration is complete and the network is converged, the routing table contains only the best route to every known destination. It must be noticed that the routing table is located in RAM memory of a router, which implies that every time the router is reloaded, the routing table is rebuild from the beginning. Below, in figure 7, we can see an example of a routing table:



```
Corp#show ip route
...
Gateway of last resort is not set

C    192.168.13.0/24 is directly connected, Serial0/1
C    192.168.14.0/24 is directly connected, FastEthernet0/0
C    192.168.15.0/24 is directly connected, Serial0/0.102
C    192.168.20.0/24 is directly connected, Serial0/0.117
R    192.168.16.0/24 [120/1] via 192.168.15.2, 00:00:05, Serial0/0.102
R    192.168.17.0/24 [120/1] via 192.168.15.2, 00:00:05, Serial0/0.102
R    192.168.30.0/24 [120/2] via 192.168.20.2, 00:00:25, Serial0/0.117
R    192.168.19.0/24 [120/1] via 192.168.20.2, 00:00:25, Serial0/0.117
R    192.168.21.0/24 [120/3] via 192.168.20.2, 00:00:25, Serial0/0.117
R    192.168.214.0/24 [120/1] via 192.168.14.2, 00:00:22, FastEthernet0/0
```

Image 3: Example of Routing Table

The first letter of a line, “C”, indicates the source of the information. The network address with its prefix comes second in the line. The numbers inside the brackets, ex [120 / 3], are arranged as [Administrative Distance / Metric of route]. Next appears the Next hop ip address, the time passed since the last update or since the router knew the route and last the router’s exit interface to reach this network.

4. STATIC ROUTING

In Static Routing, every route in a private network must be configured manually by the network administrator. The administrator must insert in each router's routing table the remote network addresses and how to forward the frames to every destination.

The implementation of static routing in a router is simple. First of all, the routing table must be checked for the current routes. If the router has no other routing technologies configured, the routing table will only include the directly connected networks. The thinking behind the static routing is simple: "Which networks the router has in its routing table? For every other network in the topology the administrator needs to configure a static route." The administrator has two options for a static route setup: a) configure the exit interface, or b) configure the next hop's ip address. A static route with an exit interface instructs the router to forwards packets for a certain network out this interface. At the other hand, a static router configured with a next hop ip address tells the router send traffic for a certain network to that specific ip. In the above topology R1's and R2's routing tables will be used as examples.

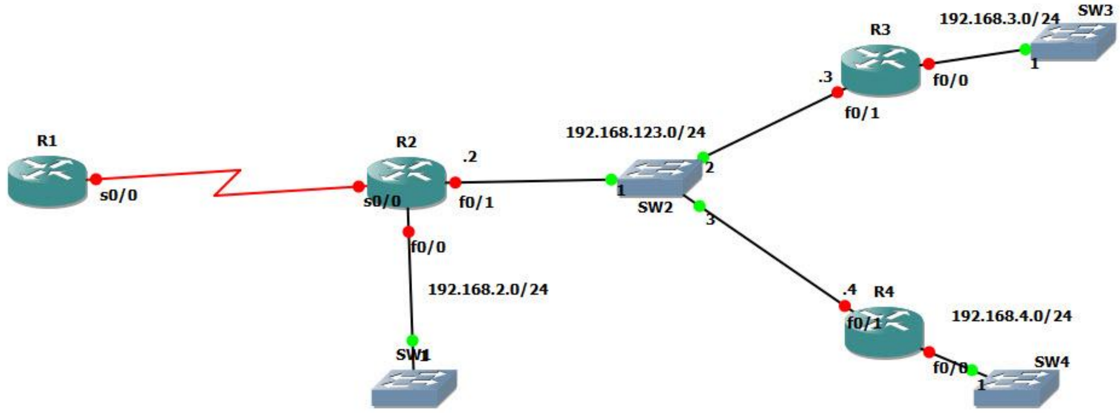


Figure 4: Static Routing Example

R1 should learn the other networks in the topology. The administrator configures a static route in R1, for every other network in the topology, using for all networks the exit interface serial0/0. The configuration will be like (the commands will probably differ from router to router but in any case the configuration follows the same pattern):

```
#ip route 192.168.2.0 255.255.255.0 serial0/0
#ip route 192.168.123.0 255.255.255.0 serial0/0
#ip route 192.168.3.0 255.255.255.0 serial0/0
#ip route 192.168.4.0 255.255.255.0 serial0/0
```

In case of R1 the use of "next hop's ip address" setup is useless because R1 will forward the traffic to R2 via their point-to-point link. The exit interface serial 0/0 must a valid interface and in "up/up" state for the static route to be shown in the routing table.

The configuration in R2 will be different because of multi-access network at its right side. The use of "exit interface" setup, at this point, cannot help R2 to forward data to the right router because the fastethernet link connects to two different routers. In this case, R2 needs to be configured with "next hop's ip" and the configuration will be like:

```
#ip route 192.168.3.0 255.255.255.0 192.168.123.3
#ip route 192.168.4.0 255.255.255.0 192.168.123.4
```

This kind of static route setup has the disadvantage that the router will have to make two lookups in the rooting table in order to find where to send a packet destined for either 192.168.3.0/24 or 192.168.4.0/24. One lookup in order to find where to forward that packet and one lookup to find which interface routes to this next hop ip. Routing this way increases the usage of CPU and Ram in the router. This phenomenon is known as "recursive table lookup".

4.1 Types of Static Routing

There are several types of static routes. All static routes are configured in the same way but they have different effect on a router's routing table [6].

- *Simple Static route:* A simple static route adds one entry in a router's routing table, instructing a router on how to forward packets destined for a certain network. For example, the static route #ip route 192.168.1.0 255.255.255.0 serial0/0/0, instructs the router to forward packets, with a destination ip in 192.168.1.0/24, out the serial0/0/0 interface.

- *Summary Static route:* A summary route is one static route which is used instead of several static routes. It is a good practice to keep small routing tables to save CPU and Ram resources and to accelerate routing. In case of stub routers, instead of several static routes used to forward traffic at networks to the same destination, a summary route may be used. For example, in R1 instead of 3 static routes for 192.168.2.0/24, 192.168.3.0/24 and 192.168.4.0/24, it can be configured one static route for 192.168.0.0/22 pointing at the same destination. This setup will have the same effect on R1's routing because it is a stub router. Be careful that summary static routes must not include networks that exist in different location than the static route is pointing.

  - *Default route:* A default route is the most used type of static route. A default route is always at the end of a router's routing table and matches any ip that has not been matched in the previous entries. In other words, any traffic that a router does not have specific entry in its routing table for it, it is send wherever the default route states. A default route appears in a routing table as 0.0.0.0/0. A default route is, also, used in stub routers to route to traffic to the core network. (Stub routers have only one connection to the rest of the network, so either they have full routing tables or they have only a default route – the result would be the same for the routing). Default route is also called the quad-zero route.

4.2 Floating Static Routes

Redundancy is always important in a network. However, set up routing redundant paths with static routing needs some attention. Suppose that we have a router with two links to a destination network. One primary leased line with a bandwidth of 10mb/s and one backup ISDN with a bandwidth of 128kb/s. Configuring the router with two static routes, one sending traffic via the 10mb/s link and one via the isdn link, will result the router to load balance packets to that destination due to the same administrative distance of the static routes. The solution to the problem is simple and it just needs to change the administrative distance of the second static route with a higher value. The Administrative distance of a static route can be changed by adding the preferred AD to the end of the static route. For example, #ip route 192.168.1.0 255.255.255.0 serial0/0/0 5. Number 5 is the new administrative distance of this static route.

Another way of configuring backup paths with static routes is by mixing a static route with tracked objects. An “object” in networking world is a predefined procedure which is repeated periodically and measures network parameters or monitors connectivity to a host and returns a True or False flag. For example, an “object” could be a permanent measure of delay in a VoIP network which returns “true” as long as the delay remains below a threshold or a periodic ping to ISP’s DNS server in order to test internet connectivity which returns “false” when some ping are lost. By tracking an object, an administrator can connect the object with a routing process. In case of static routing, an object can be combined with a static route and as long as the object returns “true” the static route will remain into the routing table. In case of a “false” return, the static route will be removed from the routing table. A static route can be combined with a tracked object by adding at the end of the command syntax the number or name of the tracking object as shown above,

#ip route 192.168.1.0 255.255.255.0 se0/0/0 track 10, where the number 10 is the number of the tracked object.

4.3 IPv6 Static Routes

IPv6 static routes are configured the same way as IPv4 static routes. The only difference is that the IPv6 prefix length of the destination network is entered rather than the dotted decimal form of the IPv4 network mask. For example, a static route in IPv6 has the form of

```
# ipv6 route 2001:db8:c18::/64 serial 0/0
```

where the /64 indicates the prefix length.

5. DYNAMIC ROUTING

5.1 RIP

Routing Information Protocol is the oldest, standards-based, interior gateway protocol used by routers to exchange routing information. Defined in *RFC 1058*, RIP version 1 was published in 1988, followed by RIP version 2 a decade later, in 1998. It is classified as a Distance-Vector protocol. Its first version uses classful routing while the second supports classless routing.

Its primitive algorithm makes the RIP suitable for small, flat, networks with few routing devices. Both versions of RIP use the Bellman-Ford algorithm in order to determine the best route to a destination. As an original distance vector protocol RIP calculates only direction and the distance to reach a network and does not store any other information about the path.

5.1.1 Metric

In order to calculate the distance to a network, RIP uses the “hop count” as a metric value. Every router in a path is considered one hop. RIP prefers the least cost path to reach the destination and does not take into measure other parameters like delay or bandwidth. That’s why RIP in networks with redundant links cannot produce reliable routing tables. For example:

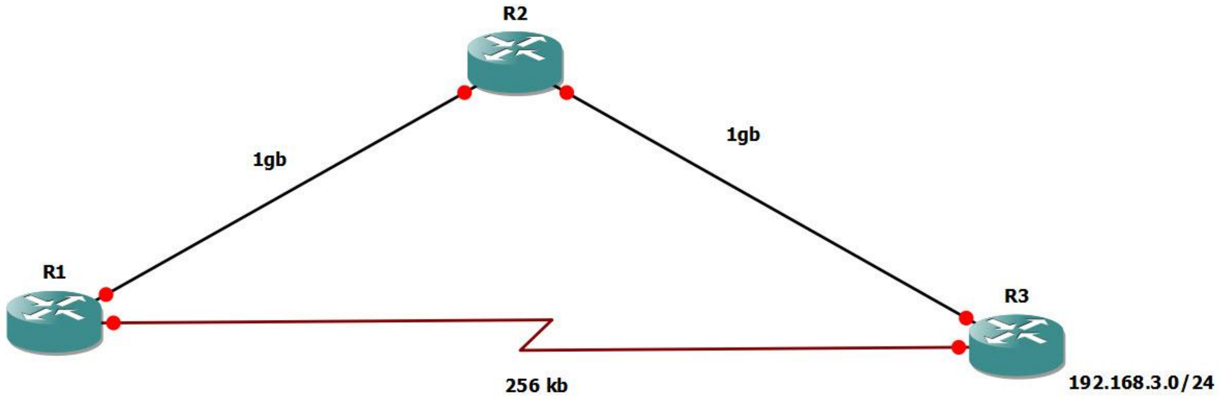


Figure 5: RIP’s hop count metric behavior

In the above topology, router R1 has two paths to reach the network 192.168.3.0/24 behind router R3. One path has two gigabit ethernet links and is two hops away and the other is a backup frame-relay circuit of 256 kbps and is one hop away. Unfortunately, RIP will choose the frame-relay circuit as the best path because it is one hop away and as a result has the lowest metric.

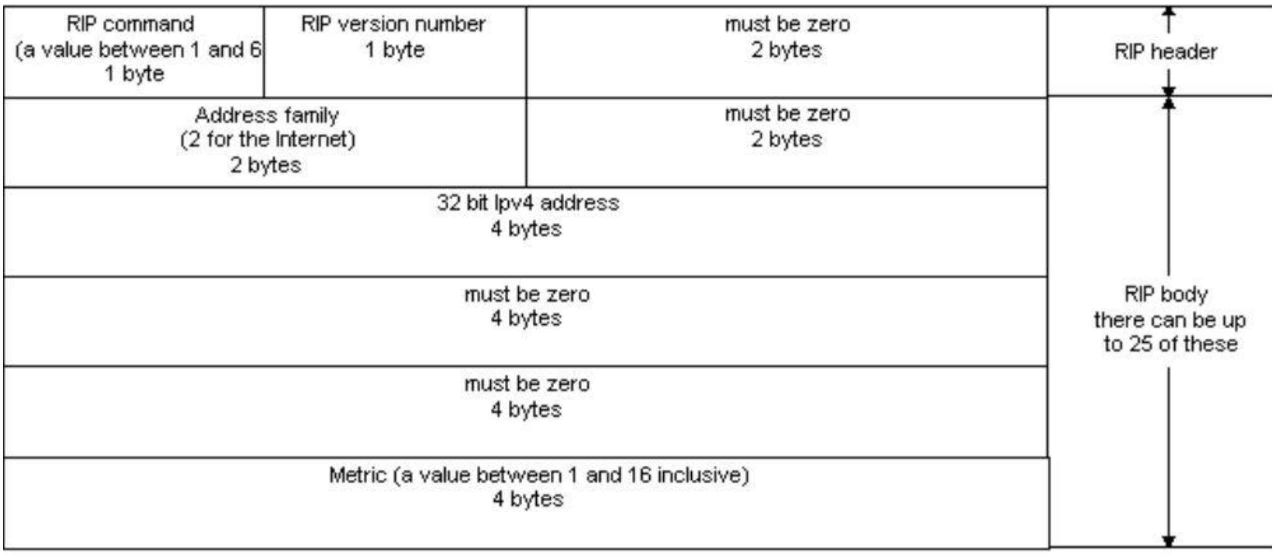
The Bellman-Ford algorithm does not prevent rooting loops from happen. Common problem in RIP was the “count to infinity”. Due to a rooting loop, a router could delete an entry in routing table and receive it again with higher metric from a different source. This can be repeated all the time and will cause a router to remove and add the same network with a higher metric, endlessly. To prevent this situation, RIP uses a maximum metric of 15 hops. A route with a metric of 16 is considered to be unreachable.

5.1.2 Packets

RIP uses two packets in order to complete its operation [2]:

- *Rip Request*: When Rip is enabled in an interface, broadcasts out a Rip Request, requesting neighbor's routing table.
- *Rip Response:* It comes either as a reply to a Rip Request including router's routing table entries or when a network change happens in the router like a metric change or a new network addition. It is commonly used to name Rip Responses as Rip Updates.

Both Rip Request and Response packets are sent as broadcast, using UDP port 520, in Rip version 1 which causes security issues in our network.



<table>
<tr><td>RIP command
(a value between 1 and 6
1 byte</td><td>RIP version number
1 byte</td><td>must be zero
2 bytes</td><td>RIP header</td></tr>
<tr><td colspan="2">Address family
(2 for the Internet)
2 bytes</td><td>must be zero
2 bytes</td><td rowspan="5">RIP body
there can be up
to 25 of these</td></tr>
<tr><td colspan="3">32 bit Ipv4 address
4 bytes</td></tr>
<tr><td colspan="3">must be zero
4 bytes</td></tr>
<tr><td colspan="3">must be zero
4 bytes</td></tr>
<tr><td colspan="3">Metric (a value between 1 and 16 inclusive)
4 bytes</td></tr>
</table>

Image 4: Rip packet format

5.1.3 Timers

Rip uses four different timers to keep routers with up to date routing information and, also, prevent routing loops [6]:

- *Update Timer:* Rip broadcasts out every Rip enabled interface a Rip Response every 30s. Because Rip was developed in times where the links were very slow and supported, mostly, half duplex communication, updates are sent with a very small variance in time, to avoid collisions between neighbor routers.
- *Invalid Timer:* If 180s pass since the last received update for a network, then the invalid timer is enabled and it sets the metric of this network to 16, marking it as unreachable.
- *Hold-down Timer:* Hold-down timer is enabled at the same time with Invalid timer to prevent routing loops. After the hold-down timer is enabled for a network, if the router receives an update with a higher metric than the last known, then the router will discard this update.

- *Flushed Timer:* The Flushed Timer is enabled after 240s since the last received update for a network and it removes the network from the routing table.

5.1.4 Loop Prevention Mechanisms

In order to prevent routing loops, due to weaknesses of Rip's algorithm, they have been developed several preventing mechanisms. These mechanisms, working all together, can offer an about 99% loop free topology with Rip. The loop prevention features are [6]:

- *Split – Horizon:* a classic feature of Distance vector protocols. When a router creates an update to send out from an interface, split – horizon tells the router not to include in this update the routes that it learned from this interface. Split – horizon can cause many issues in NBMA networks or in Hub and Spoke topologies where, most of the times, it needs to be closed in hub's interfaces. Eigrp, also, uses Split – Horizon.
- *Route Poisoning:* when a router detects that one its directly connected route has failed, it sends out an update for that network with an infinite metric (16). A router that receives the "poisoned" update knows that the route has failed and does not use it anymore.
- *Poison Reverse:* when a router receives a "poisoned" update, it advertises it back from the interface it received it. This prevents loops from happening due to synchronization of routing updates. For example, a router

5.1.5 Rip version 2

Rip version 2 (*RFC 2453*) was developed in 1998 to face Ripv1 disadvantages. Four major improvements have been added to RIPv2 to optimize its routing capabilities:

- *Classless routing:* it includes the subnet mask for each route advertised in tis updates.
- *Multicast updates:* updates are sent as multicast the ip 224.0.0.9 using UDP port 520.
- *Authentication:* support of plain text (*RFC 1723*) or md5 (*RFC1321*) authentication to secure routing updates.
- *Route tags:* allows to tag routes based on their origin.

Apart from the above enhancements Rip share the same setup and features with RIPv1.

5.1.6 Rip Next Generation (RIPng)

Rip Next Generation for IPv6 is based on RIPv2 and is defined in RFC 2080. It is not an extension of it as it is not support IPv4. RIPng uses the same timers, procedures and messages types as RIPv2. It uses the same hop-count metric, with a value of 16 indicating an unreachable network. It uses the same Request and Response messages which are being sent as multicast to FF02::9, using UDP port 521. An exception to these parallel functions is authentication. RIPng does not have an authentication mechanism of its own, but instead relies on the authentication features built into IPv6 (for example IPSec). Also, although RIPv2 encodes the next-hop ip into each route inside an update, RIPng requires specific encoding of the next hop ip for a set of route entries. At last, due to nature of IPv6, RIPng does not support Automatic Summarization like the older versions.

5.2 Enhanced Interior Gateway Routing Protocol (Eigrp)

Enhanced Interior Gateway Routing Protocol (EIGRP) is a distance vector routing protocol designed by Cisco Systems and was released in 1992. At first, it was a proprietary routing protocol, but the past year Cisco decided to make it an open protocol and requested an RFC number for it. It has adopted many features from link state protocols which makes it far more advanced than a true distance vector protocol like RIP and it is a classless routing protocol. That is why in bibliography many times is mentioned as hybrid protocol.

Eigrp uses the Diffusing Update Algorithm (DUAL) to calculate its best, loop free routes, in contrast with its predecessor, the old IGRP protocol, which was a classful routing protocol and used the old Bellman Ford Algorithm. It runs directly on top of the IP header and it has its own protocol number which is 88 [4].

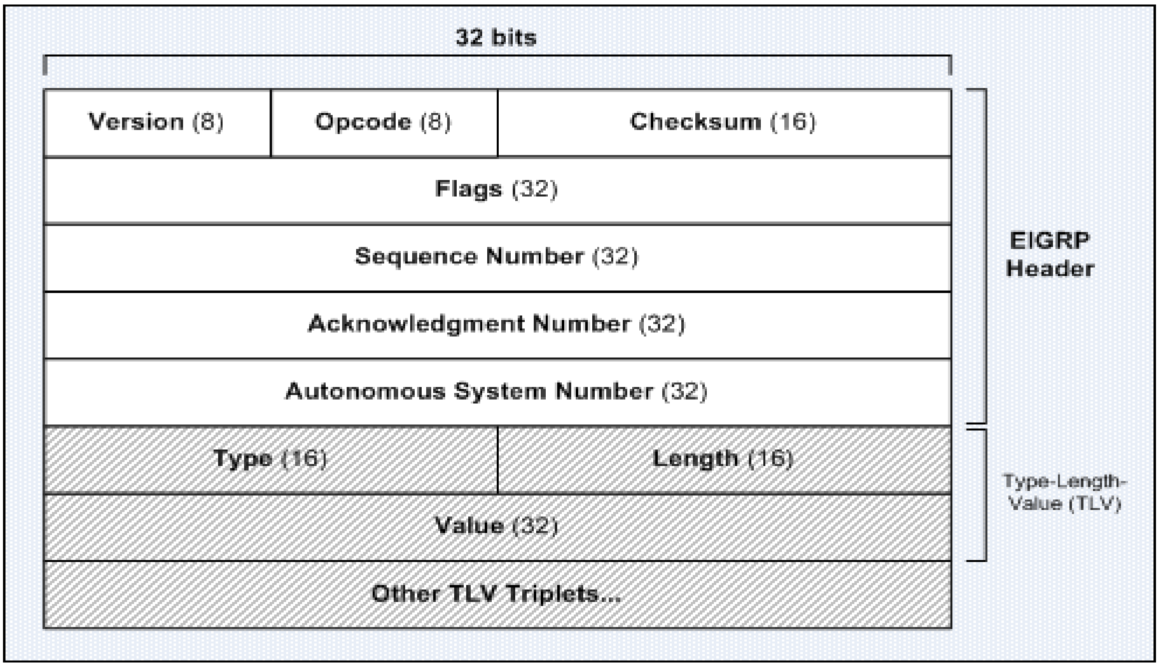


Image 5: Eigrp Header Packet

5.2.1 Eigrp Packets

Eigrp uses five types of packets to complete its purpose. These packets are sent either multicast, using the multicast ip address 224.0.0.10, to a group of routers or as unicast. It was developed to be independent from TCP/UDP protocols so it uses *Reliable Transport Protocol (RTP)* for the delivery of its packets. RTP can send packets either reliable requesting an acknowledgement from the receiver or unreliable. The Eigrp's five packets are listed below [2]:

- *Hello:* hello packets in eigrp, as in any other modern routing protocol, contain all the required parameters which must match in order for two routers to become neighbors or "adjacent" in a routing protocol and they have dual usage. When an interface is enabled in Eigrp, hellos are sent for neighbor discovery. Once, the neighborship has been established, hellos are sent periodically, every 5 or 60 seconds to maintain the neighborship.

- *Update:* contains routing information and is sent reliable to each eigrp neighbor. When the adjacency comes up, routers in Eigrp exchange their full routing tables and after that, they send an update only if a change happens. Updates in Eigrp are partial and bounded. Partial means that they contain only the information needed by the neighbor, for example the addition of a new network, and not the entire routing table. Bounded means that the update would be sent only to routers that are affected by the change.

- *Query:* used when an Eigrp router has lost information about a certain network and doesn't have any backup paths in the topology table. The router will send query packets to its directly connected neighbors asking them if they have information about this particular network.

- *Reply:* used as a reply to a query. The number of reply packets received must be the same as the number of queries sent. If a router does not receive all the replies, either positive or negative, the Eigrp algorithm "stuck" in Active state and so it cannot forward any traffic to the requested network. This problem is known as "Stuck In Active" (SIA) problem in Eigrp.

- *Acknowledgment:* Always sent using a unicast address to confirm the reception of an Eigrp packet.

5.2.2 Eigrp Router-id

Each time the Eigrp starts in a router, it must choose a Router-id (RID). The RID is a unique number, in x.x.x.x format, looks like an IP address, and it is used to uniquely identify an Eigrp process running on a router inside a routing domain. Each router in order to choose its RID follows the process:

- Checks if it is manually configured by the administrator, under routing protocol's setup.

- If not, then, the RID will be the highest IP address of the loopback interfaces.

- If there is no loopback interface, then, the RID will be the highest interface IP of the router. The interface is not mandatory to be in Eigrp.

Eigrp RID is not needed in common Eigrp configuration and daily tasks. It is used when external routes are redistributed into Eigrp, RID is checked. A router cannot accept external routes from a router with the same RID.

5.2.3 Eigrp Timers

Eigrp uses two different timers whose values vary based on the underlying network and are listed below [6]:

- *Hello:* Interval at which Eigrp sends hello messages on an interface.

- *Hold:* Timer used to determine when a neighbor router or interface is down, based on a router not receiving any Eigrp hello messages.

Table 3: Eigrp Timers

	Point-to-Point / Multiaccess	**NBMA**
Hello	5s	60s
Hold	15s	180s

To optimize convergence, a network administrator can simply reduce the Hello and Hold timers, accepting insignificant additional overhead, in return for faster convergence times. Changes to timers can be made either per interface or per Eigrp process. Timers in Eigrp don’t have to match between neighbors.

5.2.4 Eigrp Tables

Eigrp supports multiple Network layer protocols like IPv4, IPv6, IPX etc. For this purpose, it uses Protocol – Dependent Modules (PDMs), one for each protocol, in order to keep them separately. Its PDM contains three tables for Eigrp which are:

- *Neighbor Table:* it stores all the information about the directly connected routers with which our router has form an adjacency in Eigrp.

- *Topology Table:* after two routers have become neighbors, they will exchange their routing information which is stored in the Topology Table. At the end of the Eigrp’s process, topology table will contain the best routes to every destination and every backup route which fulfills the Feasibility Condition.

- *Routing Table:* the best routes from the topology table are added in the routing table together with any other routes that the router has.

All three tables are stored in RAM and every time the router reboots or the administrator resets them - they are rebuilt from the beginning.

5.2.5 Eigrp Neighborship Requirements

A router running Eigrp cannot form a neighbor relationship with any router. It checks the follow values to ensure that the requesting router is eligible to become its neighbor or not [2]:

- Active Hello packet on the link
- Neighbors must be in the same subnet
- Neighbor must pass authentication
- Same AS Number in the hello packets
- Same K – values enabled

5.2.6 Eigrp Metric

Eigrp uses a composite metric which can be associated with six variables but it can be computed only by four of them [6]. These variables are:

- *Bandwidth (K1):* The minimum bandwidth (in kb/s) across the path from router to the destination network.
- *Load (K2):* Integer in range 1 to 255, computed in 5 minutes intervals. 255 means that the link is full.
- *Delay (K3):* The sum of delays, in 10s of microseconds, across the path from router to destination network.
- *Reliability (K4):* Integer in range 255 to 1, computed in 5 minutes intervals. 1 means that the link is totally unreliable.
- *MTU (K5):* Refers to the minimum MTU of the path from router to the destination network. MTU is not used in metric calculation. It is used in case we have more than 8 equal cost paths to a destination in order to exclude paths.
- *Hop count:* Hop count is the hop count reported by the next-hop router and is used only to limit the diameter of the network. By default, the maximum hop count is set to 100 and can be configured from 1 to 255. If the maximum hop count is exceeded, the route will be marked as unreachable by setting the delay value to 0xFFFFFF.

Eigrp uses the below formula to calculate the metric for a network:

*Eigrp metric = 256 * [(K1*Bw) + (K2*Bw) / (256 – Load) + K3 * Delay)] * [K5 / (Reliability + K4)]*

The values K1 through K5 are configurable weights. The default values are K1=K3=1 and K2=K4=K5=0 and can be changed. The K values K2 and K4 equal to zero because load and reliability are dynamic values and they can change over time. We do not want Eigrp routers calculating 24/7 and sending updates to each other each time one of these two changes. By abstracting the zero values the formula can be rewritten to:

Eigrp metric = Bw_metric + Delay_metric where,

*Bw_metric = (10^7 / min_bw_of_the_path) * 256 and*

*Delay_metric = (sum_of_delays_of_the_path / 10) * 256*

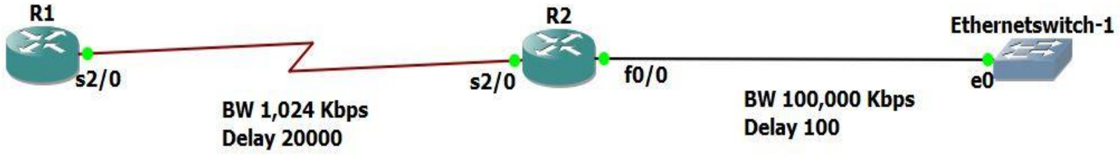


Figure 6: Topology Example 2

In the above topology (Figure: 6), total metric calculation to R2’s lan will be:

*Bandwidth metric = (10,000,000/1024) * 256 = 2,499,840*

*Delay metric = (20100/10) * 256 = 514560*

R1’s Eigrp metric to R2’s lan = Bandwidth + delay = 3014400

5.2.7 Dual Algorithm

Eigrp uses Diffusing Update Algorithm to calculate the best routes to remote networks. The design philosophy behind DUAL is that even temporary routing loops are detrimental to the performance of a network. DUAL uses diffusing computations to perform distributed shortest-path routing while maintaining a loop free topology. After the initial exchange of hellos and the neighbor adjacencies, Eigrp neighbors start to exchange their routing updates. The first routing updates contain all routes known by the sending routers and the metrics of those routes. Any further update, if any, includes any other change in the network. For each router, the router will calculate a distance based on the distance advertised be the neighbor and the cost of the link to that neighbor.

Neighbor's best metric to a destination is called *Reported Distance (RD)* of that destination. Our router's lowest calculated metric to each destination will become the *Feasible Distance (FD)* of that destination. The neighbor which provides the best path to a destination becomes the *Successor* for that destination. The *Feasibility Condition (FC)* is a condition that is met when the neighbor's RD to a destination is lower than the router's FD to that same destination. The fulfillment of FC is a basic requirement for a route to be saved as a backup in the topology table. The neighbor whose routes meet the FC become the *Feasible Successor* for that routes. Because Feasible successors always have a smaller metric to a destination, a router will never choose a path that will lead back through itself, creating a loop.

When an Eigrp router is performing no diffusing computations, each route is in the *Passive State*. In order the router to use a route it must be in passive state. A router will reassess its list of feasible successors for a router any time an input event occurs. Such an event can be [2]:

- A change in the metric or the state of a directly connected link.
- The reception of an update packet.
- The reception of a query or reply packet.

The first step in its recalculation is a local computation in which the distance to the destination is recalculated for all feasible successors. While the router is performing a local computation, the router remains in the passive state. If a feasible successor exists, an update will be sent to all neighbors. If a feasible successor is not found, then, the router will begin a diffusing computation and the router will change to the active state. During the computation, the router cannot change anything about that route and cannot forward traffic to it. The diffusing computation starts by sending queries to all of its neighbors. Each neighbor, upon receipt of the query, will either send a reply to the router if it has a successor for this destination, or it will change the route to active state and will begin a diffusing computation too. For each neighbor to which a query is sent, the router keeps track of all outstanding queries. The diffusing computation is complete when the router has received a reply to every query sent to every neighbor.

5.2.8 Eigrp over NBMA Networks

Eigrp uses the multicast address 224.0.0.10 to discover new neighbors and sent/receive updates. However, NBMA networks, by default, do not forward multicast and broadcast traffic through them. In order to address these problems, two solutions exist:

- Manually configure the neighbor in the router under the Eigrp setup. This configuration sent hellos as unicast to the configurable neighbor.
- Whenever it is possible, change the default behavior of NBMA networks and allow multicast and broadcast traffic through them.

5.2.9 Wildcard mask

Eigrp has also adopted the use of the wildcard mask instead of the use of the known subnet mask. While in a subnet mask the 0 means ignore and 1 means match – in wildcard mask we have the opposite behavior, 0 means match and 1 means ignore. A wildcard mask is produced from a certain subnet mask with the above calculation:

  255.255.255.255

- 255.255.255.0

= 0.0.0.255

Or

  255.255.255.255

- 255.255.252.0

= 0.0.3.255

From the 255.255.255.255 we abstract the subnet mask and the result is the wildcard mask. Eigrp configuration requires the use of wildcard mask

The basic difference between the two is that with subnet mask we can match specific networks but with a wildcard mask we can match any ip.

5.2.10 Authentication

Eigrp uses authentication to authenticate every Eigrp message. The routers use the same pre-shared key (PSK), calculating an MD5 digest for each Eigrp message based on that shared PSK. If a router configured for Eigrp authentication receives an Eigrp message that does not pass the authentication checking based on the local copy of the key, then the router drops the message. Two routers cannot become neighbors if the authentication between them fails.

Eigrp authentication prevents unauthorized routers from becoming neighbors and protect the router from DoS attacks. However, Eigrp authentication does not prevent unauthorized users from read the messages. Authentication configuration in Eigrp is a two-step process: First, we create a Key-Chain with all the available keys and then, we configure the Eigrp enabled interfaces to authenticate each message using the specified Key-chain and MD5.

For example, the above script configures one key-chain named TEST with two keys and enable eigrp authentication on the Gigabit 0/0 interface.

#key-chain TEST

#key 1

#key-string password123

#key 2

#key-string di123

```
#interface gigabit 0/0
#ip authentication key-chain eigrp 10 TEST
#ip authentication mode eigrp 10 md5
```

5.2.11 Features

In order to improve its functionality and optimize its routing behavior, Eigrp has been enhanced with some features which run either by default or can be enabled manually. These features are [2]:

- *Unequal load balancing:* Eigrp can be configured to load balance traffic between unequal metric routes. This can be achieved by configuring the *variance* parameter. *Variance* is a multiplier which shows the maximum metric that a route should have in order to enter the routing table for load balance. The equation is

  Max_metric_load_balance = variance x best_metric_for_route

  and variance by default is set to 1.

- *Stub Router:* Stub router is a feature for improving network stability and conserve CPU and Ram resources. It is used on spokes in Hub-and-Spoke topologies to prevent sub-optimal routing and to eliminate Eigrp query storms in the network. When enabled to a spoke router, the router advertises only the specified routes to the hub router. The router will not advertise routes received from other Eigrp neighbors to the hub router and, also, other routers do not forward queries towards the spoke.

- *Nonstop Forwarding (NSF) Awareness:* The Eigrp NSF Awareness feature allows an NSF-Aware router that is running Eigrp to forward packets along routes known to a router performing a switchover operation. This allows Eigrp neighbors of the failing router to retain the routing information that it has advertised and to continue using this information until the failed router resumes normal operation and is able to exchange routing information. The Neighborship is maintained throughout the entire NSF operation.

- *Split-Horizon:* Split-horizon feature is also enabled in Eigrp. When an Eigrp enabled router "learns" a network from a specific interface, it does not include this network to the update that it sends out this interface. Needs caution at hub-and-spoke topologies where it needs to be disabled in the hub router.

5.2.12 Eigrp over IPv6

For Eigrp was easy to add support for IPv6 due to its architecture. By using PDMs, Eigrp creates separate tables to work with IPv6. The configuration of Eigrp for IPv6 (or Eigrpv6) and IPv4 is similar as many of the same processes and operational functionality are the same in both versions. However, there are a few differences that are listed below:

- Eigrp for IPv6 uses the neighbor's link local address for neighbor discovery, neighbor adjacency and as the next hop IP address.

- Eigrp for IPv6 does not require neighbors to be in the same IPv6 subnet in order to become neighbors.

- Updates are sent as multicast to the multicast address FF02::10.

- It supports only md5 authentication and IPv6's built-in mechanisms.

- Both versions use a 32-bit number for Router-ID. However, in the IPv6 version it has to be configured manually.

- Eigrp for IPv6 does not support automatic summarization.

5.3 Open Shortest Path First (Ospf)

Open Shortest Path First (OSPF) is a link-state protocol which was firstly developed in 1987 by the *Internet Engineering Task Force (IETF)* OSPF Working Group. In 1989, the specification for OSPFv1 was published in RFC 1131. At the beginning, were two versions of Ospf; one to run on routers, and another one to run on UNIX systems. The later implementation, later, became the known UNIX process, GATED. The first version of Ospf was an experimental routing protocol that was never deployed in daily networks. In 1991, the improved Ospf version 2 was introduced in RFC 1247 by John Moy. In 1998, the Ospf version 2 was updated in *RFC 2328* and is the current usable version for IPv4 networks. Last but not least, in 1999, OSPF version 3 was published in *RFC 2740* in order for Ospf to support IPv6 networks. During the last years there have been numerous of RFCs containing updates of OSPF, like *RFC 1584*, *1587* and *1850*.

OSPF is a link-state protocol which creates a loop free topology using Dijkstra's Shortest Path First (SPF) algorithm to create an SPF Tree. Each Ospf router maintains a link-state database containing the routing information received from all other routers. When the database is complete, the router runs Dijkstra's algorithm on it to create the SPF tree. The SPF tree is then used to populate the IP routing table with the best paths to each network.

5.3.1 Packets

OSPF uses five different types of Link-State Packets (LSPs) to serve its purpose. Each packet serves a specific purpose in the Ospf routing process [7]:

- *Hello:* used to discover new Ospf neighbors and advertise them the parameters on which the routers must agree to become neighbors. Hello packets are, also, used to accomplish the DR/BDR Election process on multi-access networks.

Some of the parameters included in each hello packet are: router-id, area id, network mask, hello and dead intervals, router's priority, lan's DR/BDR and a list of router's neighbors. Hellos are sent as multicast using using the address 224.0.0.5 .

- *Database Description (DBD):* it contains an abbreviated list of the sending router's link-state database and is checked from the received routers against their local link state-database.

- *Link-State Request (LSR):* When a router receives a DBD packet it compares it with its local link-state database. For every unknown network, the router sends a link-state request to request more info about the specific network.

- *Link-State Update (LSU):* are used either as replies to LSRs or to announce changes to neighbors. Link-state updates contain eleven different types of Link-State Advertisements (LSA) which are used to advertise different type of routing information.

- *Link-State Acknowledgement (LSAck):* When a router receives a LSU, the router responds back with a link-state acknowledgment to confirm the receipt of the LSU.

5.3.2 Link State Advertisements

Link-state updates contain different types of Link-state Advertisements. Each Advertisement is produced from different source and for different purposes. When a router receives an LSU, copies the LSA in its local link-state database (LSDB) and, then, it forwards it to all Ospf neighbors it has. It's a good practice to keep the LSDB as small as possible in order to improve convergence time and to consume less router resources. The list below contains all LSAs and their use [2]:

- *LSA Type 1 – Router LSA:* it contains information about the directly connected networks, the cost to them and the neighbors a router has within the area that it belongs. Router LSAs are flood within the area the router belongs until the ABR. All routers inside an area have the same LSA type 1 in their LSDBs. The link-state ID of LSA type 1 is the originating router ID.

- *LSA Type 2 – Network LSA:* it is created by Designated Router (DR) for multi-access networks in order to describe itself and the routers that are connected to that segment. Network LSAs are flooded across their own area and the link-state ID is the IP interface address of the DR.

- *LSA Type 3 – Summary LSA:* it is created by an Area Border Router (ABR) in order to advertise the networks from one Ospf area to another. In order to reduce the size of LSDBs, LSA type 3 does not contain full details about the networks of one are but only a summary of them.

- *LSA Type 4 – ASBR Summary LSA:* it is used to announce an Autonomous-System Border Router (ASBR) to an area and it is generated by an ABR. It contains the router ID of the ASBR.

- *LSA Type 5 – External LSA:* it is generated by an ASBR to import networks from other routing protocols and is flooded throughout the entire Ospf Autonomous System. During the propagation of LSA type 5, the advertising router ID is not changed. By default, all routers in the Ospf domain will not calculate the cost to the ASBR but they will accept the cost that the ASBR added to them when redistributed these networks in the AS. These routes are known as Ospf External – Type 2 routes and appear as "O E2" in router's routing table. This behavior can change manually and the routes will turn to Ospf External – Type 1 routes, "O E1". External – Type 1 routes will appear with different metric in each router because now each router adds its own cost to the ASBR in addition to the already redistributed metric.
- *LSA Type 6 – Multicast LSA:* is a specialized LSA that is used in multicast Ospf (MOSPF) applications. It is defined as Group Membership LSA but it has been obsolete.

- *LSA Type 7 – NSSA External LSA:* it is generated by an ASBR to propagate external routes inside a Not-So-Stubby area (NSSA). By default, external networks (LSA 5) are not allowed inside a NSSA. LSA type 7 is a "trick" to allow a NSSA area to have its own ASBR. LSA type 7 is translated in LSA type 5 in order to be advertised outside a NSSA area by the local ABR. NSSA External LSAs are appeared in routing tables as "O N1" and "O N2" corresponding to "O E1" and "O E2".

- *LSA Type 8 – External Attributes LSA for BGP:* A link-local only LSA for OSPFv3. LSA type 8 is used to give information about link-local addresses and a list of IPv6 addresses on the link. In OSPFv2, was originally intended to be used as External-Attributes-LSA for transit autonomous systems where OSPFv2 could replace the internal Border Gateway Protocol (iBGP). In these networks, the BGP destinations would be carried in LSA type 5 while their BGP attributes would be inserted into LSA type 8. Most OSPFv2 implementations never supported this feature, and it was never standardized for OSPFv2.

- *LSA Type 9,10,11 – "Opaque" LSAs:* these LSAs are mostly designed to extend the capabilities of Ospf and to allow for transmission of arbitrary data that Ospf does not need to care about. Each one of these three LSAs has a different scope of flooding:

- *LSA Type 9:* Link-Local scope, defined in RFC2370
- *LSA Type 10:* Area-Local scope, defined in RFC2370
- *LSA Type 11:* AS scope, defined in RFC5250

5.3.3 LSA Flooding

After the adjacencies are established, the router might begin sending out LSAs. As the flooding term implies, the advertisements are sent to every neighbor [7]. In turn, each received LSA packet is copied and forwarded to every neighbor except the one that sent the LSA. This process is the key advantage of a link state protocol over a distance vector. LSA's are forwarded almost immediately, without any processing, and as a result they converge much faster than the distance vector protocols.

Ospf refloods each LSA every 30 minutes based on each LSA's Age variable. The router that creates the LSA set this age to 0 seconds. Each router that receives the LSA increments the age of its copy of each LSA over time. If 30 minutes pass with no changes to an LSA, meaning that no other reason existed in that 30 minutes to cause a reflooding of the LSA, the owning router increments the sequence number, reset the timer to 0 and refloods the LSA. Ospf flushes its LSAs after 60 minutes.

5.3.4 Timers

Ospf uses two different timers whose values vary based on the network type and are listed below:

- *Hello:* Interval at which Ospf sends hello messages on an interface.
- *Hold:* Timer used to determine when a neighbor router or interface is down, based on a router not receiving any Ospf hello messages.

Table 4: Ospf Timers

	Point-to-Point / Broadcast	**NBMA**
Hello	10s	30s
Dead	40s	120s

To optimize convergence, a network administrator can simply reduce the Hello and Dead timers, accepting insignificant additional overhead, in return for faster convergence times. Changes to timers can be made per interface. Timers in Ospf must match between neighbors.

5.3.5 Router-id

Each time the Ospf starts in a router, it must choose a Router-id (RID). The RID is a 32-bit unique number, in x.x.x.x format, looks like an IP address, and it is used to uniquely identify an Ospf process running on a router inside a routing domain. Each router in order to choose its RID follows the process [7]:

- Checks if it is manually configured by the administrator, under routing protocol's setup.

- If not, then, the RID will be the highest IP address of any loopback interface.

- If there is no loopback interface, then, the RID will be the highest interface IP of the router. The interface is not mandatory to be in Ospf.

Ospf RID is has important role in the Ospf process. It is written in LSAs to determine the LSA's originator router. Also, sometimes, is used in DR/BDR election in Ospf broadcast network type. In order to change router-id, you have to restart Ospf process or to reload the router. A router's RID must be unique in the network, otherwise it will not form an adjacency or accept updates from a router with the same RID.

5.3.6 Metric Calculation

Ospf uses the SPF (Shortest Path First) algorithm to select the best route for the routing table. It is also referred as Dijkstra algorithm. Ospf uses a parameter called "cost" to calculate the metric for a route. Cost is the inverse proportional of bandwidth of an interface and is calculated by the following formula [6]:

Cost = Reference bandwidth / Interface bandwidth in bps,

Reference bandwidth was defined as arbitrary value in Ospf documentation (RFC 2338). Vendors need to use their own reference bandwidth. Most of them use the 100Mbps bandwidth as reference. Also, the cost of an interface can be set explicitly. The total metric for a route is cumulative cost of all outgoing interfaces in a route.

5.3.7 Router Types

There are four types-roles of Ospf routers which are determined by a router's function or location within the Ospf domain. A router may have more than one role from the above [7]:

- *Internal (IR):* a router with all of its interfaces belonging to same Ospf area.
- *Backbone (BR):* a router with at least one interface to the backbone area. The backbone area is the area 0. A backbone router may also be an ABR.
- *Area Border Router (ABR):* a router which has one interface, at least, in two different areas. The ABR router is responsible for the routing information exchange between the different areas.
- *Autonomous-System Boundary Router (ASBR):* a router which imports routing information from other routing sources into the OSPF domain.

5.3.8 Network Types

OSPF has different behavior when running over different network topology types. The network type for OSPF depends on data link layer protocol that runs on the interface. There are three main network types defined in OSPF, (RFC 2328), and two more industry build [2]:

- *Point-to-Point:* a topology where two routers are directly connected via a single link like, for example, a serial link using PPP. All OSPF packets are multicast to 224.0.0.5.

- *Broadcast:* OSPF uses it in a multi-access segment, where many devices can be connecting with each other, like a switch connecting five or more routers. In a broadcast segment, DR/BDR election takes place to reduce the amount of LSAs in the network. As mentioned before, when a router receives LSA it forwards it to all its neighbors. In a broadcast segment, this behavior can lead to a huge amount of the same LSAs spreading across the multi access network cause network overload.

- *Non-Broadcast Multi-Access (NBMA):* Not all multi-access technologies support broadcast transmissions. Protocols like Frame-Relay, ATM or X.25 are the most common examples of non-broadcast transport. Every router on the segment must configured with the IPv4 address of each Ospf neighbor manually. Ospf hello packets are then sent individually transmitted as unicast packets to each adjacent neighbor. A DR/BDR election also takes place to limit the number of adjacencies formed.

- *Multipoint Non-Broadcast:* Unlike an NBMA topology, this network type it seeks to organize the virtual circuits into a collection of point-to-point networks. Hello packets must still be replicated and transmitted individually to each neighbor manually but without the need of DR/BDR election.

- *Point-to-Multipoint:* Ospf treats point-to-multipoint networks as a collection of many point-to-point links. DR/BDR election is not needed and neighbor adjacencies can be achieved automatically via hello packets.

- *Loopback interfaces:* By default, virtual loopback interfaces on a router are treated as a stub host via Ospf and are advertised as /32 networks.

5.3.9 DR/BDR Election Process

In multi-access networks Ospf has to overcome two problems relating to the flooding of LSAs [6]. First of all, the formation of an adjacency between every attached router would create many unnecessary LSAs in the segment. If *n* is the number of routers on a multiaccess network, there would be *n(n-1)/2* adjacencies. Each router would flood *n-1* LSAs for its adjacent neighbors, plus one for the network, resulting in n^2 LSAs originating from the network. Also, the operation of flooding on the network itself would create a mess. A router would flood an LSA to all its adjacent neighbors, which in turn would flood it to all their adjacent neighbors, creating many copies of the same LSA on the same network. To prevent these problems, a Designated Router (DR) and a Backup Designated Router (BDR) are elected on the multi-access segment. The DR has the following responsibilities:

- To represent the multi-access network and its attached routers to the rest of the Ospf area.
- To manage the flooding process on the multi-access network.

Each router on the network forms a full adjacency only with DR and the BDR. With every other router in the segment the adjacency state stays to 2-way level. A BDR is elected in addition to the DR in order to minimize the network unavailability if the DR fails. If this happens, the BDR instantly takes over the DR role. The DR/BDR election criteria are the following in sequence:

- *The router with the highest priority becomes the DR*. Each multi-access interface of a router has a *Router Priority,* which is an 8-bit unsigned integer ranging from 0 to 255. Different routers have different default values of priority. A router with a priority of 0 does not participate to the election process.
- If the election based on priority comes to a draw, then *the router with the highest Router-ID wins the election.*

Every other router on the multi-access network becomes a *DR-other*. All DR-other router uses the ip 224.0.0.6 to multicast their LSU to DR and BDR. Next, DR will forward the LSU to every other neighbor using the multicast ip 224.0.0.5.

It must be noticed that if a router enters a network where there is already a DR and a BDR elected, it will accept them and will not triggered a new election. In addition, a router running Ospf can have multiple roles, for example a router might be a DR on one if its attached networks and a BDR or DR-other in another interface.

5.3.10 Neighbor State Machine

An Ospf router transitions a neighbor through several states before the neighbor is considered fully adjacent [7]:

- *Down:* The initial state of a neighbor indicates that no Hellos have been heard from the neighbor in the last dead-interval timer. If neighbor transitions to the Down state from some higher state, the link state Retransmission, Database Summary and Link State Request lists are cleared out.
- *Attempt:* This state applies only to neighbors on NBMA networks, where neighbors are manually configured. A DR-eligible router transitions a neighbor to the Attempt state when the interface to the neighbor first becomes Active or when the router is the a DR/BDR.
- *Init:* This state indicates that a Hello packet has been seen from the neighbor in the last dead-interval, buy a two-way communication has not been established yet. The router includes the RIDs of all neighbors in this or higher state, inside the Neighbor field of the Hello Packets.

- *2-Way:* This state indicates that the router has seen its own RID in the Neighbor field of the neighbor's Hello packets, which means that a bidirectional conversation has been established.
- *ExStart:* In this state, the router and its neighbor establish a master/slave relationship and determine the initial DD sequence number in preparation for the exchange of Database Description packets. The neighbor with the highest RID becomes the master.
- *Exchange:* The router sends Database Description packets describing its entire link-state database to neighbors that are in the Exchange state. The router may also send Link State Request packets, requesting more recent LSAs to neighbors in this state.
- *Loading:* The router sends Link State Request packets to neighbors that are in the Loading state, requesting more recent LSAs that have been discovered in the Exchange state but have not yet been released.
- *Full:* Neighbors in this state are fully adjacent, and the adjacencies appear in the Router and Network LSAs.

5.3.11 Ospf Areas and Area Types

In Ospf, a single autonomous system (AS) can be divided into smaller sub-domains called *Areas*. An area is a logical connection of Ospf networks, routers and links that have the same area identification. The concept of areas reduces the number of Link-State Advertisements and other Ospf overhead traffic sent on the network, and, also, it reduces the size of topology database that each router must maintain. It is recommended to design an area as a collection of contiguous IP subnetted networks.

A router within an area must maintain a topological database for the area to which it belongs. The router does not have detailed information about network topology outside of its area, which thereby reduces the size of its database. Areas limit the scope of route information distribution. It is not possible to do route update filtering within an area. However, route summarization and filtering is possible between different areas. The Link-State Database (LSDB) of routers within the same area must be synchronized and be exactly the same. The reduced size of a database reduces the impact on a router's memory, CPU utilization and the LSA flooding scope. Areas are identified by an area ID. Area ID can expressed either in decimal format or in IP address format (ex. Area 0 or Area 0.0.0.0).

Each Ospf multi-area network must follow these rules:

- A backbone area, which combines a set of independent areas into a single domain, must exist. Area 0 is considered to be there the backbone area.
- Each non-backbone area must be directly connected, either by physical link or by virtual link, to the backbone area.
- The backbone area must not be partitioned under any failure condition.

Using the area design approach there are three types of traffic that may be defined in relation to areas:

- *Intra-area:* traffic consists of packets that are passed between routers within a single are.
- *Inter-area:* traffic consists of packets that are passed between routers in different areas.
- *External:* traffic consists of packets that are passed between a router within the Ospf domain and a router in another routing domain. External traffic is divided

into two different sub-categories: 1) External-type 1 (O E1) and 2) External-type 2 (O E2).

Ospf areas are created to control and minimize the movement of LSAs across the Ospf network. Thus, there are six different types of areas:

- *Standard:* Any area with normal treatment in LSAs. LSA Type 1 and 2 are sent between routers within an area. LSA Type 3 and 5 are used to share inter-area and external routes to this area.

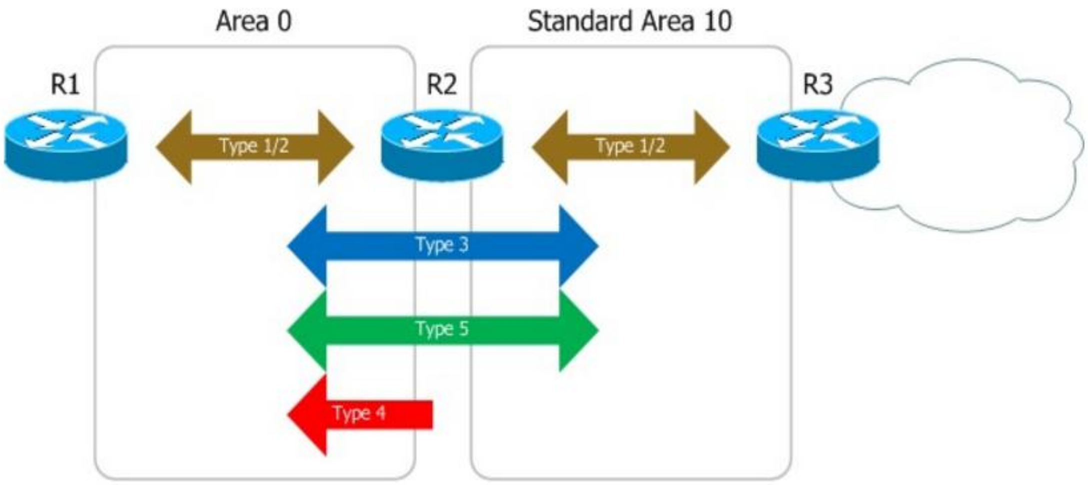


Figure 7: LSA Exchange in Multi-Area Ospf

- *Backbone:* Backbone area is a standard area with the prerequisite that all other areas must be connected to this area. Backbone is always the Area 0.

- *Stub:* In a Stub area, the LSA Type 5 (External LSA) are filtered and replaced with a default route generated by the ABR. This ensures that routers inside the stub area will be able to route traffic to external destination without having to maintain all the individual external routes.

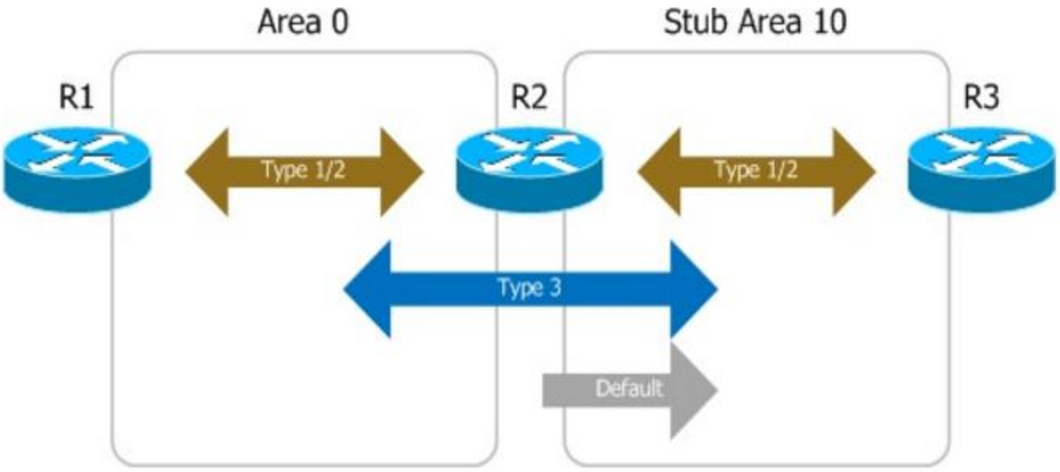


Figure 8: Ospf Stub Area

- *Totally Stubby:* Totally Stub areas do not receive LSA Type 3,4,5 from their ABRs. All routing out of the area relies on a single default route injected by the ABR.

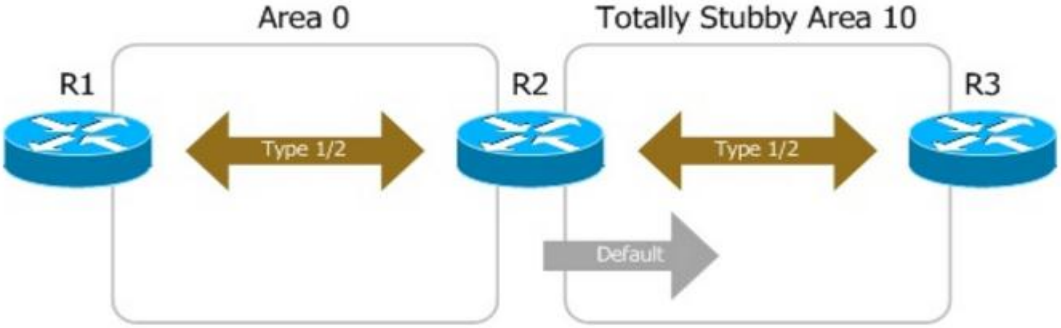


Figure 9: Ospf Totally Stubby Area

- *Not-so-Stubby (NSSA):* Stub and totally reduce the resource utilization of routers but neither type can contain an ASBR. An NSSA area filters LSA Type 4, 5 and uses LSA Type 7 to allow a router to advertise external routes inside the area. The ABR converts the LSA Type 7 to LSA Type 5 before flooding them to the rest Ospf domain. Unlike the others, the ABR will not inject a default route into an NSSA area unless it is manually configured to do so.

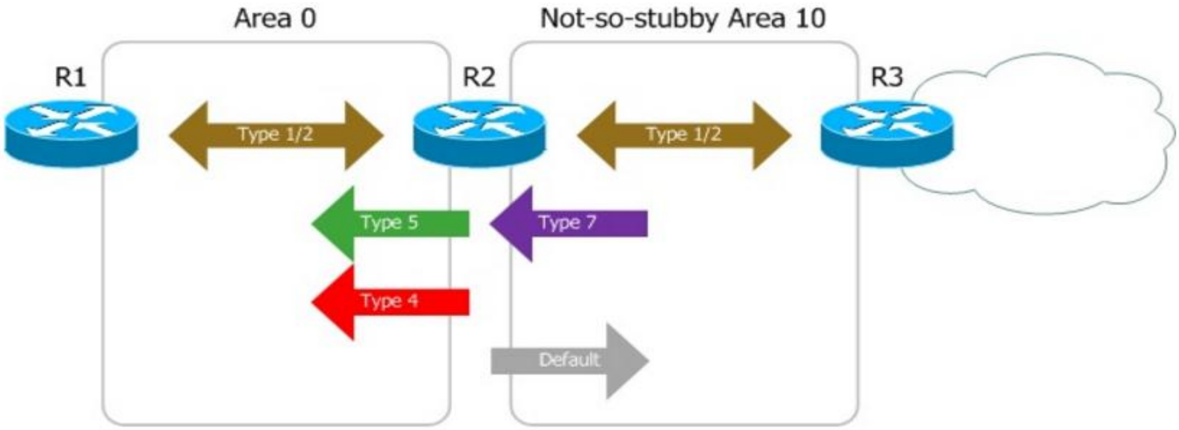


Figure 10: Ospf NSSA Area

- *Totally Not-so-Stubby:* A Totally NSSA area filters LSA Type 3, 4, 5 and uses LSA Type 7 to allow a router to advertise external routes inside the area. The ABR converts the LSA Type 7 to LSA Type 5 before flooding them to the rest Ospf domain. Unlike the NSSA area, the ABR will automatically inject a default route into the area.

Over any type of stub area, it cannot be implemented any Ospf Virtual link.

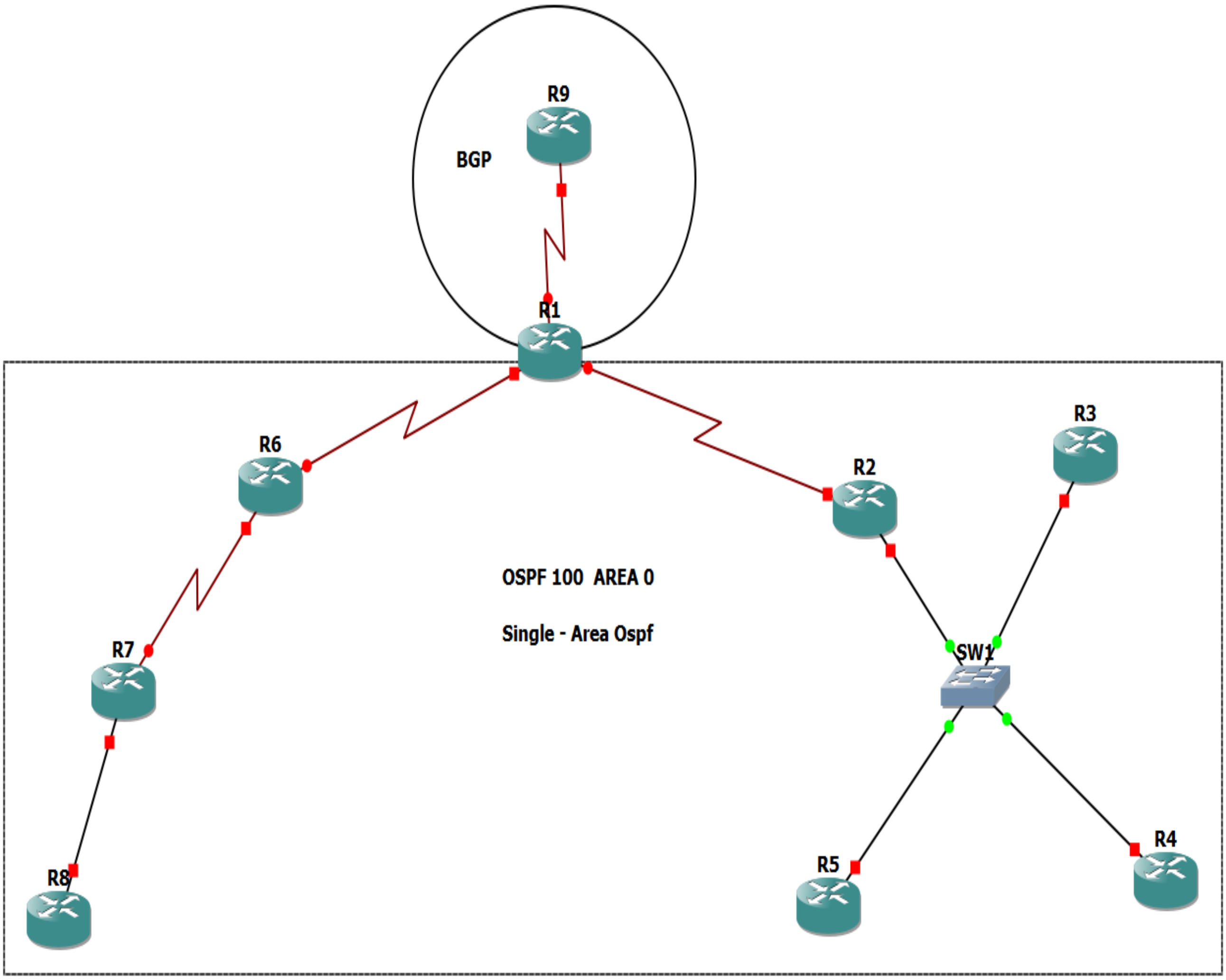


Figure 11: Ospf Single-Area Topology

At figure 11, all routers are belonging to area 0. As a result, they must have identical databases. On the other hand, in the next figure (Figure 12) the same physical topology is implemented using Multi-Area Ospf. Routing domains have separated in several areas. Routes inside an area have identical databases but for the other areas turns to be a distance-vector logic.

However, the above topology will experience routing inconsistencies because Area 20 has no router with an interface in Area 0. R8 will exchange and receive only routing information about Area 20. In order to overcome this problem, Ospf has a built-in mechanism, called *Virtual-Link*, which is configured the two ABRs, R7 and R6. Virtual link connects, logically, Area 20 to Area 0 in order to receive full routing information about the other areas.

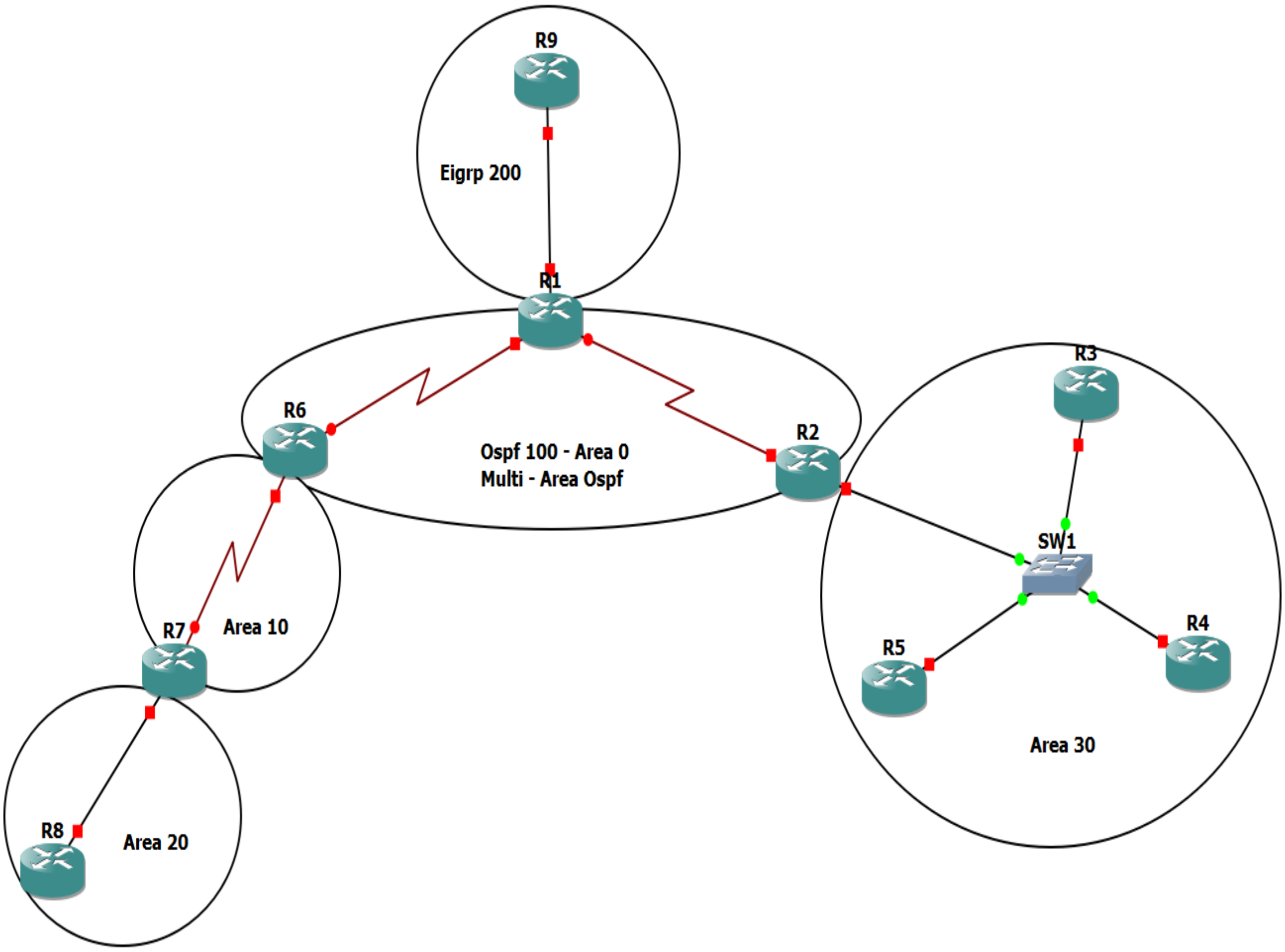


Figure 12: Ospf Multi-Area Topology

5.3.12 Authentication

Ospf has the capability of authenticating all packets exchanged between neighbors. Authentication causes routers to authenticate every Ospf message exchanged. Routers use the same pre-shared key value, generating an MD5 digest for each Ospf message and sending that digest as part of each Ospf message. If a router configured for Ospf authentication receives an Ospf message, and the received message's MD5 digest does not pass the authentication checking based on the local key value, the router silently discards the message. As a result, an adjacency can not be formed because the router ignores inauthentic Ospf Hello packets. Ospf supports three types of authentication [7], where the third is the most preferable option in production networks:

- *Type 0:* No authentication
- *Type 1:* Plain text
- *Type 2:* MD5 cryptographic checksums

5.3.13 Neighborship Requirements

A router running Ospf cannot form a neighbor relationship with any router. It checks the follow values to ensure that the requesting router is eligible to become its neighbor or not:

- Unique Ospf RID

- Same Hello and Dead Intervals
- Neighbors must be in the same subnet
- Same Network Type
- Same Area ID
- Neighbor must pass authentication
- Same Area Type
- Same IP MTU value between neighbors

5.3.14 Ospf over IPv6 (OSPFv3)

OSPFv3 was first specified in *RFC 2740* and was updated later in *RFC 5340*. OSPFv3 is not only a new protocol implementation but also a major rewrite of the internals of the protocol [5]. Like RIPng and EIGRP for IPv6, the processes and operations of OSPFv3 are almost the same as the fundamentals mechanisms of OSPFv2. The following is a comparison of the main features of OSPFv2 and OSPFv3:

- OSPFv3 uses the same metric, timers, areas, LSAs and packets as OSPFv2.
- OSPFv3 advertises IPv6 routes instead of IPv4.
- Both versions, use the same hello mechanism to learn about neighboring routers and form adjacencies. However, in OSPFv3, there is no need for the neighbors to be in the same subnet because adjacencies, now, are formed using the link local addresses and not the global unicast addresses.
- OSPFv3 uses the multicast addresses FF02::5 and FF02::6.
- In OSPFv3, messages are sourced using the link address of the exit interface.
- OSPFv3 uses only the authentication and encryption features provided by IPSec.
- Both versions, use the same 32-bit Router ID.

- OSPFv3 supports of multiple instances per link by adding an Instance ID to the OSPF packet header.

5.4 Intermediate System to Intermediate System (IS-IS)

IS-IS is a link state IGP routing protocol. It was developed in the late 1980s by Digital Equipment Corporation (DECnet Phase V) and was standardized by the International Standards Organization (ISO) in ISO/IEC 10589. The current version of this standard is ISO/IEC 10589:2002 [8].

The purpose of IS-IS was to make possible the routing of datagrams using the ISO-developed OSI protocol stack called CLNS. ISO/IEC 10589 defines support for the ISO *Connectionless Network Protocol (CLNP)* as defined in ISO 8473. However, the protocol was designed to be extensible to other network protocols. In the RFC 1195 was defined IS-IS support for IPv4, and additional IETF extensions have defined IS-IS support for IPv6. Integration of support for multiple network layer protocols has led to the term *Integrated IS-IS* or *Dual IS-IS*. Typically, IS-IS is used in Large ISPs because it scales very well to very large networks.

5.4.1 Routing Domain

An IS-IS routing domain is a network in which all the routers run the Integrated IS-IS routing protocol to support intradomain exchange of routing information [8]. The network can consist of only IP design, only ISO CLNP design, or can contain both. The IS-IS protocol was originally intended to support only CLNP, but RFC 1195 added the support of IP protocol. The following implementation requirements are specified by *RFC 1195*:

- *Only-IP*: domain route only IP traffic but, also, support forwarding and processing of OSI packets required for IS-IS operation.

- *Only-ISO*: domain carry only ISO traffic including those required for IS-IS operation.

- *Dual protocol:* domain routes both IP and OSI CLNP traffic simultaneously.

It is also possible to design a dual domain so that some areas route IP only, whereas others route CLNP only, and yet others route both IP and CLNP. RFC 1195 imposes restrictions on the manner in which IP and CLNP can be mixed within an area. The underlying goal is to achieve consistent routing information within an area by having identical Level 1 link-state databases on all routers in that area. All routers inside an area must be configured in the same way, either for IP-only or CLNP-only or both. A router is not allowed to have a set of links dedicated to IP only and another set to CLNP and yet another set to both protocols. However, at the domain level, there is no restriction on mixing areas that are uniformly IP-only with other areas that are uniformly CLNP-only or uniformly configured for both IP and CLNP. In order words, all links inside an area must be configured the same way, but links in the backbone area can have the attached routers configured differently.

A routing domain can be one single domain or divided into many subdomains. In IS-IS each subdomain is referred as an area and is assigned an area address. Routing within an area is referred as *Level-1* routing. Routing between Level-1 areas is referred as *Level-2* routing. A router (called an Intermediate System (IS)) can operate at Level 1, Level 2, or both. ISs that operate at Level 1 exchange routing information with other Level-1 ISs in the same area and maintain a database with all routers in the area. ISs that operate at Level 2 exchange routing information with other Level-2 routers regardless of whether they are in the same Level-1 area. Level 1-only routers are aware of the local area topology only, which involves all the nodes in the area, the next-hop routers to reach them and default routes to the Level-2 routers. Level 1 routers depend on Level 2 routers for access to other areas and all other destinations outside the area. The set of Level-2 routers and the links that interconnect them form a Level-2 subdomain, which must not be partitioned in order for routing to work properly**.** Level-2 router maintains two separate databases: one Level-1 database for intra-area routing and a Level-2 database for inter-area routing.

5.4.2 Addressing

As a protocol originally designed for routing CLNP packets, IS-IS uses a node-based addressing scheme of CLNP as its basic addressing premise. Integrated IS-IS, which can be used for routing IP packets, inherits many concepts from the original specification, including the CLNS addressing scheme for identifying network nodes. However, IS-IS uses ISO network addresses. Each address identifies a point of connection to the network, such as a router interface, and is called a *Network Service Access Point (NSAP)*. An end system can have multiple NSAP addresses, in which case the addresses differ only by the last byte (called the n-selector). Each NSAP represents a service that is available at that node.

Unlike other IP routing protocols, which typically run on TCP, UDP, or IP, which are OSI Network or Transport Layer protocols, IS-IS runs directly on the Data link layer. As a result, an interface that runs IS-IS doesn’t need an IP address to exchange IS-IS information. Instead, only the router needs an IP address. An IS-IS address is called Network Entity Title (*NET)*. While an IP address is 32 bits long and normally written in dotted quad notation (such as 192.168.1.2), a NET can be 8 to 20 bytes long (usually is 10 bytes long). An example of a NET address is the address 49.0001.1921.6800.1002.00.

The IS-IS address consists of three parts [8]:

- *Area Identifier:* The first three bytes are the Area ID. The first byte is called *Address Family Identifies (AFI)* of the authority, which is equivalent to the IP address range assigned to an AS. RFC 1918 specifies that AFI value 49 is what IS-IS uses for private addressing. The second two bytes of the Area ID represent the IS-Is area number.

- *System Identifier:* The next six bytes identify the node (the router) on the network. The system identifier is equivalent to the host portion on an IP address. Although, we can choose any value for the system identifier, a commonly used method is to use binary-coded decimal (BCD). This involves taking the router’s IP address, filling in all leading zeros, and then repositioning the decimal points to form three two-byte numbers. For example, if we pad the IP address 192.168.1.2 with zeros, the result is 192.168.001.002. After rearranging the decimal points, the address is 1921.6800.1002. Another way to assign the system identifier is to use the router's mac-address, which is a six-byte address. For example, a router's mac-address of 00:1b:63:31:86:be, can be used as a system identifier like 001b.6331.86be.

- *NET Selector:* The final two bytes are the NET selector (NSEL). For IS-IS, they must always be 00, to indicate the current system.

5.4.3 PDU Types

ISs exchange routing information with their peers using Protocol Data Units (PDUs). Each PDU is identified by a five-bit type number. IS-IS uses the following types of PDUs [8]:

- *IS-IS Hello PDUs (IIHs)*: Hello packets are used to establish and maintain adjacencies between IS-IS neighbors. IIHS include the system ID of the sender, the assigned area address and the identity of neighbors on a circuit that are known to the sending IS. There are three types of IIHS:

  - *Point-to-Point*: IIHs used on point-to-point links (PDU Type 17).

  - *Level-1*: Sent on multi-access links when the sending IS operates as a Level-1 router on that link (PDU Type 15).

  - *Level-2*: Sent on multi-access links when the sending IS operates as a Level-2 router on that link (PDU Type 16).

- *Link-State PDUs*: LSPs are used to distribute routing information between IS-IS nodes. An IS router, also, generates an LSP to advertise its neighbors and the destination that are directly connected to the IS. Each link-state PDU must be refreshed periodically on the network and is acknowledged by information within a sequence number PDU. There are two types of LSPs:

  - *Level-1*: generated by all Level-1 ISs (PDU Type 18) and are flooded throughout the Level-1 area.

  - *Level-2*: generated by Level-2 ISs (PDU Type 20) and are flooded only to the Level-2 subdomain.

- *Sequence Number PDUs*: used to control distribution of LSPs and to provide mechanisms for synchronization of the distributed Link-State Databases on the routers in an IS-IS routing area. They contain a summary description of one or more LSPs. There are two types of SNPs for either Level-1 or Level-2:

  - *Complete Sequence Number PDUs*: they contain a complete list of all link-state PDUs in the IS-IS database. CSNPs are sent periodically on all links. The receiving routers use the information in the CSNP to update and synchronize their link-state PDU databases. The designated router multicasts CSNPs on broadcast links in place of sending explicit acknowledgments for each link-state PDU. Contained within the CSNP is a link-state PDU identifier, a lifetime, a sequence number, and a checksum for each entry in the database. Periodically, a CSNP is sent on both broadcast and point-to-point links to maintain a correct database. The advertisement of CSNPs occurs, also, when an adjacency is formed with another router. When the device receives a CSNP, it checks the database entries against its own local Link-State Database. If it detects missing information, the device requests specific link-state PDU details using a Partial Sequence Number PDU (PSNP). (PDU Type 24 & 25). Periodically, an IS multicast Complete SNP (CSNP) to describe all the LSPs in the Pseudonode database. L1 CSNPs are sent to all Level-1 ISs multicast address 01-80-C2-00-00-14, while L2 CSNPs are sent to all Level-2 ISs multicast address 01-80-C2-00-00-15.

- *Partial Sequence Number PDUs*: used by an IS router to request Link-State PDU information from a neighboring router. It can also explicitly acknowledge the receipt of a link-state PDU on a point-to-point link. (PDU Type 26 & 27)

5.4.4 IS-IS Supported Circuit Types

IS-IS supports two generic circuit types:

- *Point-to-Point circuits:* A point-to-point circuit has exactly two ISs on the circuit. An IS forms a single adjacency to the other IS on the point-to-point circuit. The adjacency type describes what level or levels are supported on that circuit. If both ISs support Level-1 on that circuit and the ISs are configured with at least one matching address, then, the adjacency supports Level-1 and level-1 LSPs and SNPs will be sent on that circuit. If both ISs support Level- 2 on that circuit, the adjacency supports Level-2 and level-2 LSPs and SNPs will be sent on that circuit. The adjacency can be Level-1, Level-2, or Level 1 and 2. ISs send point-to-point IIHs on point-to-point circuits.
- *Multiaccess circuit:* Multiaccess circuits support multiple ISs connected in the same multiaccess network. For example, two or more operating on the circuit. The ability to address multiple systems utilizing a multicast or broadcast address is assumed. An IS that supports Level-1 on a multiaccess circuit sends Level-1 LAN IIHs on the circuit. An IS that supports Level 2 on a multiaccess circuit sends Level-2 LAN IIHs on the circuit. ISs form separate adjacencies for each level with neighbor ISs on the circuit. An IS will form a Level-1 adjacency with other ISs that support Level 1 on the circuit and will have a matching area address. It is a misconfiguration to have two ISs with disjoint sets of area addresses supporting Level 1 on the same multiaccess circuit. An IS will form a Level-2 adjacency with other ISs that support Level 2 on the circuit.

5.4.5 IS-IS Election of the Designated Intermediate System

If an IS would advertise all of its adjacencies on a multiaccess circuit in its LSPs, the total number of advertisements required would be N^2—where N is the number of ISs that operate at a given level on the circuit. To address this scalability issue, IS-IS defines a pseudo node to represent the multiaccess circuit. All ISs that operate on the circuit at a given level elect one of the ISs to act as the Designated Intermediate System (DIS) on that circuit. A DIS is elected for each level that is active on the circuit. The DIS is responsible for issuing pseudo node LSPs. The pseudo node LSPs include neighbor advertisements for all of the ISs that operate on that circuit. All ISs that operate on the circuit (including the DIS) provide a neighbor advertisement to the pseudo node in their non-pseudo node LSPs and do not advertise any of their neighbors on the multiaccess circuit. In this way the total number of advertisements required varies as a function of N—the number of ISs that operate on the circuit.

A pseudo node LSP is uniquely classified by the following identifiers:

- System ID of the DIS that generated the LSP.
- One pseudo node ID.
- An LSP number (0 to 255).
- A 32-bit sequence number.

The DIS election process is quite simple. Every IS router interface is assigned both an L1 and L2 priority in the range of 0 to 127. The IS with the highest priority will become the DIS. In a tie, the router whose attached interface has the numerically highest Mac-Address will become the DIS. The nonzero pseudo node ID is what differentiates a pseudo node LSP from a non-pseudo node LSP and is chosen by the DIS to be unique among any other LAN circuits for which it is also the DIS at this level. The DIS is also responsible for sending periodic CSNPs on the circuit. This provides a complete summary description of the current contents of the LSPDB on the DIS. Other ISs on the circuit can then perform the following activities:

- Flood LSPs that they have that are absent from or are newer than those that are described in the CSNPs sent by the DIS.
- Request an LSP by sending a PSNP for LSPs that are described in the CSNPs sent by the DIS that are absent from the local database or older than what is described in the CSNP set. In this way, the LSPDBs of all ISs on a multiaccess circuit are efficiently and reliably synchronized.

5.4.6 LSPDB Synchronization

Operation of IS-IS requires a reliable and efficient process to synchronize the LSPDBs on each IS. In IS-IS, this process is called the *Update process*. The update process operates independently at each supported level. LSPs may be locally generated, in which case they always are new LSPs. LSPs may also be received from a neighbor on a circuit, in which case they may be generated by some other IS or may be a copy of an LSP generated by the local IS. Received LSPs may be older, the same age, or newer than the current contents of the local LSPDB:

- ➢ *Handling of Newer LSPs:* A newer LSP is added to the local LSPDB. If an older copy of the same LSP currently exists in the LSPDB, it is replaced. The newer LSP is marked to be sent on all circuits on which the IS currently has an adjacency in the UP state at the level associated with the newer LSP—excluding the circuit on which the newer LSP was received. On point-to-point circuits, the newer LSP will be flooded periodically until the neighbor acknowledges its receipt by sending a PSNP or by sending an LSP that is the same or newer than the LSP being flooded. On multiaccess networks, the IS will flood the newer LSP once. The IS examines the set of CNSPs that are sent periodically by the DIS for the multiaccess circuit. If the local LSPDB contains one or more LSPs that are newer than what is described in the CSNP set (this includes LSPs that are absent from the CSNP set) those LSPs are reflooded over the multiaccess circuit. If the local LSPDB contains one or more LSPs that are older than what is described in the CSNP set (this includes LSPs described in the CSNP set that are absent from the local LSPDB), a PSNP is sent on the multiaccess network with descriptions of the LSPs that require updating. The DIS for the multiaccess circuit responds by sending the requested LSPs.

- *Handling of Older LSPs*: An IS may receive an LSP or SNP that is older than the copy in the local LSPDB. The IS marks the LSP in the local database to be flooded on the circuit on which the older LSP or SNP that contained the older LSP was received. At this point, the actions taken are identical to the actions that are described in the “Handling of Newer LSPs” section after a new LSP has been added to the local database.

- *Handling LSPs That are the Same*: Because of the distributed nature of the update process, it is possible than an IS may receive copies of an LSP that is the same as the current contents of the local LSPDB. On a point-to-point circuit, receipt of such an LSP is ignored. Periodic transmission of a CSNP set by the DIS for that circuit will serve as an implicit acknowledgement to the sender that the LSP has been received. In a multiaccess circuit, receipt of such an LSP is ignored. Periodic transmission of a CSNP set by the DIS for that circuit will serve as an implicit acknowledgement to the sender that the LSP has been received.

5.4.7 Shortest Path Calculation

Once the *Update process* has built the link-state database, the *Decision process* uses the information in the database to calculate a shortest-path tree. The process uses the shortest-path tree to construct a forwarding database. Separate SPF calculations are run for the L1 and L2 database. *ISO10589* specifies the following metrics (one required and three optional) to be used by Is-Is to calculate the shortest path [8]:

- **Default**: This metric must be supported and understood by every IS.
- **Delay**: This optional metric reflects the transit delay of a subnetwork.
- **Expense**: This optional metric reflects the monetary cost of using the subnetwork.
- **Error**: This optional metric reflects the residual error probability of the subnetwork, similar to Eigrp’s reliability metric.

Each metric is expressed as an integer between 0 and 63 and a separate route is calculated for each metric. As a result, if a system supports all four metrics, the SPF calculation must be run four times for both the L1 and L2 database. Total metric of a route is a simple sum of the individual metrics at each outgoing interface, and the maximum metric value that can be assigned to any route is 1023. This small maximum is frequently pointed out as a limitation of IS-IS because it leaves little room for metric granularity in large networks. However, newer extensions to IS-IS allow for much larger metric space (called wide metrics) of 32 bits.IS-IS classifies, also, routes as internal or external. Internal routes are paths to destinations within the IS-IS routing domain, and external routes are paths to destinations external to the routing domain. Whereas L2 routes may be either internal or external, L1 routes are normally internal. Using routing policy, a L1 route can be external, but this should be done with good justification and special care.

Given multiple possible routes to a particular destination, a L1 path is preferred an L2 path. Within each level, a path that supports the optional metrics is preferred over a path that supports only the default metric. Within each level of metric support, the path with the lowest metric is preferred.

5.5 Border Gateway Protocol (BGP)

The Border Gateway Protocol (BGP) is a robust and scalable routing protocol developed, in early 90s, to exchange routing information between different *Autonomous Systems*. Internet Service providers (ISP) and customer networks usually use an IGP protocol, such as OSPF, for the exchange of routing information within their private networks. Any communication between these IGPs and the Internet or between service providers will be accomplished through BGP.

The first ancestor to BGP was the *Gateway to Gateway Protocol (GGP)*. Briefly described in RFC 823, GGP was the first routing protocol which ran the Internet. It was a more advanced version of RIP using also a distance vector logic. In 1984, GGP gave his place to *Exterior Gateway Protocol (EGP, RFC 904)*. EGP used also distance vector logic. A limitation of EGP was that was only allowed a tree-like network topology, which means that there could be a single path between any two points in the network. EGP introduced the notion of *"Autonomous Systems"*, with each separate network engaging in EGP having its own autonomous system (AS) number. In 1989, the *Border Gateway Protocol (BGP, RFC 1105)* was introduced as successor to EGP. The first version of BGP supported only hierarchical network topologies with only up, down and horizontal relationships. That limitation was removed in *BGP version 2* a year later (*RFC 1163*). Another year later, *BGP version 3* (*RFC 1267*) came around, which improved some of BGP's inner workings.

In the early 1990s, the internet was growing quickly, with more and more networks requiring a range of IP addresses. Back in those days only classful routing was supported, and the routing tables started to grow very fast and quickly becoming too large for the routers of the day. In 1994, *BGP version 4* (*RFC 1654, RFC 1771* and *RFC 4271*) solved this issue with the addition of the notion of *Classless Inter-Domain Routing (CIDR, RFC 1519)*, which removes the notion of having the IP address space separated into three classes. This means for example that 32 continuous /24 networks in a routing table can be replaced with 1 summarized /19 network, reducing routing table's size. BGPv4 is the version that still being used today. During these years, many features have been added to BGP like, the support of communities (*RFC1997*), the multiprotocol support (*RFC 2283*), TCP session with MD5 hash (*RFC 2385*), flap damping (*RFC 2439*), route refresh (*RFC 2918*) and the support of 32-bit AS numbers (*RFC 4893*).

5.5.1 BGP Basics

Border Gateway Protocol advertises, learns and chooses the best paths inside the global Internet. When two Internet Service Providers connect, they typically use BGP to exchange routing information. Enterprises, also, use BGP to exchange routing information with one or more ISPs, allowing their edge routers to learn Internet routes. One key difference when comparing BGP to IGP routing protocols is that BGP's robust best path algorithm. This algorithm gives BGP the freedom to choose the best paths using complex rules and different settings which offers great flexibility to network engineers.

5.5.2 Autonomous System Number

An *Autonomous system (AS)* is a collection of connected IP routing prefixes under the control of a single administrative entity or domain in the Internet. Originally, the definition required control by a single entity, usually an ISP or a very large organization or institute with independent connections to multiple networks, that adhere to a single and clearly defined routing policy, as originally defined in *RFC 1771*. The latest definition in RFC 1930 came into use because multiple organizations can run BGP using private AS numbers to an ISP that connect those organizations to the Internet. Even though, there may be multiple autonomous systems supported by the ISP, the Internet only sees the policy if the ISP. That ISP must have an officially registered autonomous system number (ASN). In the beginning, AS numbers were defined as 16-bit integers, which allowed about 65,536 assignments. *RFC 4893* and *RFC 6793* introduced the use 32-bit AS numbers which offers about 4.294.967.296 AS numbers. The 16-bit AS numbers were categorized as:

0: Reserved

1 – 64.495: Public AS numbers

64.496 – 65.511: Reserved to use in documentation

64.512 – 65.534: Private AS numbers

65.535: Reserved

Private ASNs allow the routers inside an AS to participate in BGP, while using the same ASN as many other organizations.

5.5.3 Routing Towards Internet

Enterprise edge routers typically have two options for outbound routing towards Internet. Either default routing or BGP. A router with only one connection available to an ISP needs to have only one default route to its routing table. However, one or many routers with multiple connections to one or more ISPs must run BGP between the Enterprise network and the ISPs to determine the best path to a destination. There are several cases where the use of BGP is mandatory:

- Customers connected to more than one service provider.
- ISP networks themselves acting as transit systems and forwarding external traffic.
- Exchange points, which can be defined by the network access point (NAP) between a region and core.
- Very large enterprises using BGP as their core routing protocol.

Considering an Enterprise's Internet connectivity options, the available solutions can be grouped to four separate cases:

- Single Homed (1 link per ISP, 1 ISP)
- Dual Homed (2 or more links per ISP, 1 ISP)
- Single Multihomed (1 link per ISP, 2+ ISPs)
- Dual Multihomed (2 or more links per ISP, 2 or more ISPs)

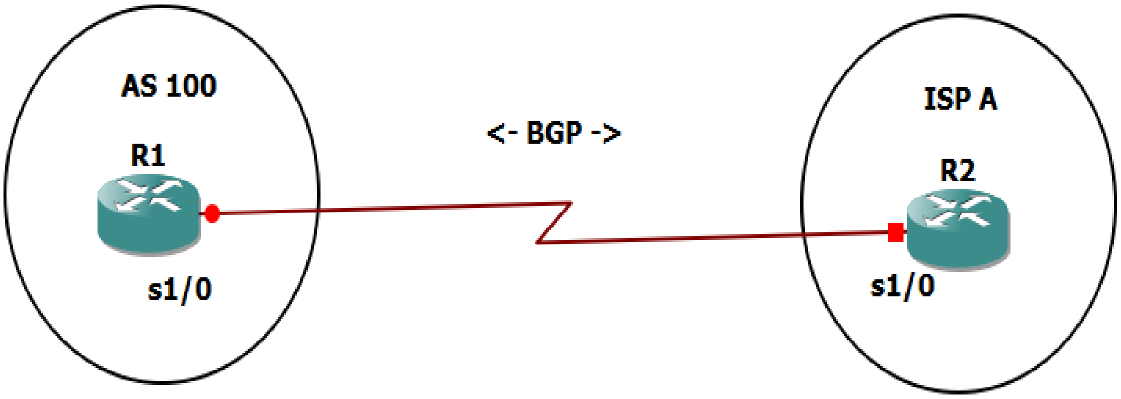


Figure 13: Single Homed Topology

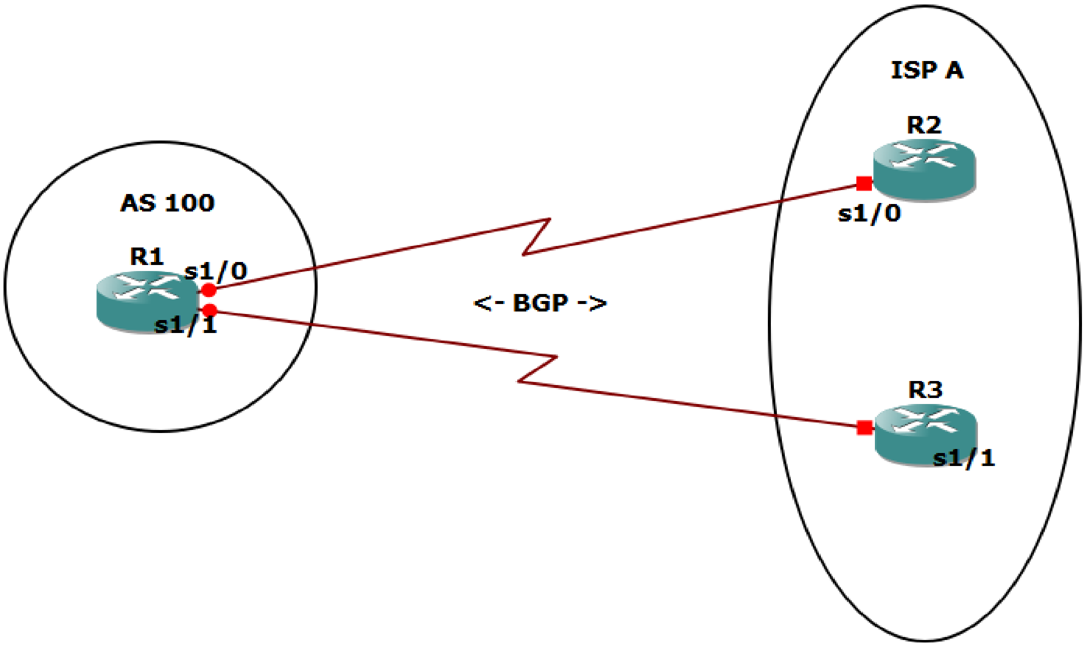


Figure 14: Dual Homed Topology

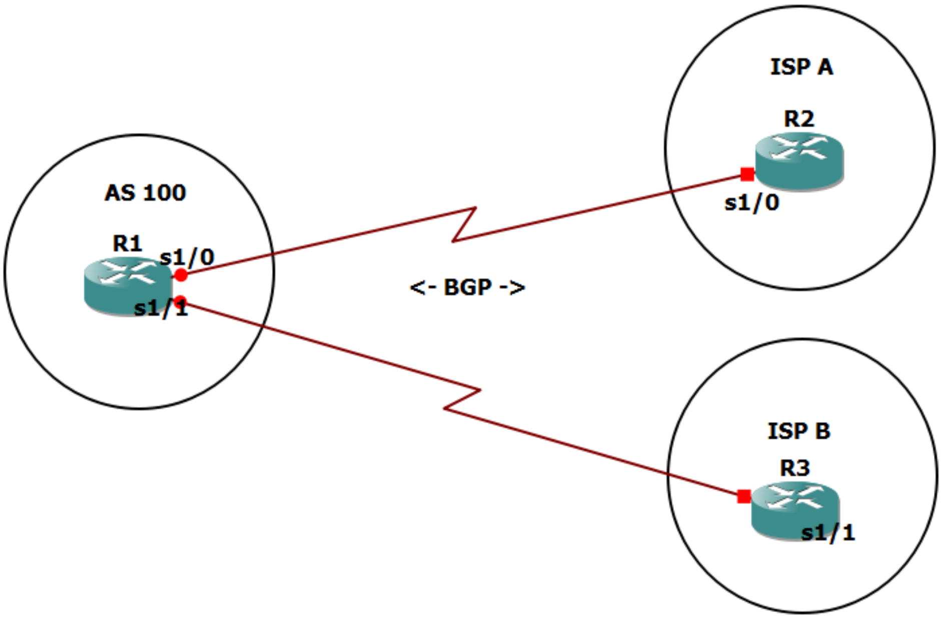


Figure 15: Single Multihomed Topology

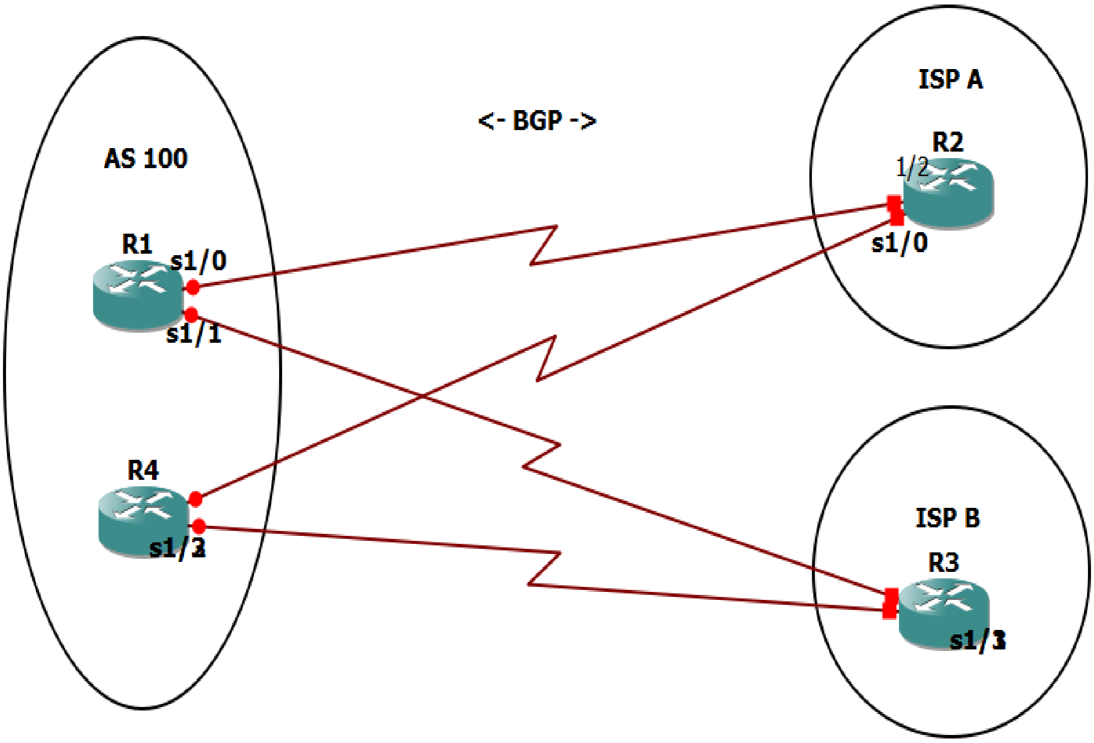


Figure 16: Dual Multihomed Topology

5.5.4 BGP Peers

Unlike the other IGP protocols, BGP does not have neighbor detection capability. BGP neighbors are called peers and must be configured manually. A router configured to run BGP is called a BGP “speaker”. A BGP speaker connects to another speaker, either in the same or different AS, by using a TCP connection to port 179, allowing peering routers to be on the same subnet or several routers away. The TCP connection is maintained throughout the peering session.

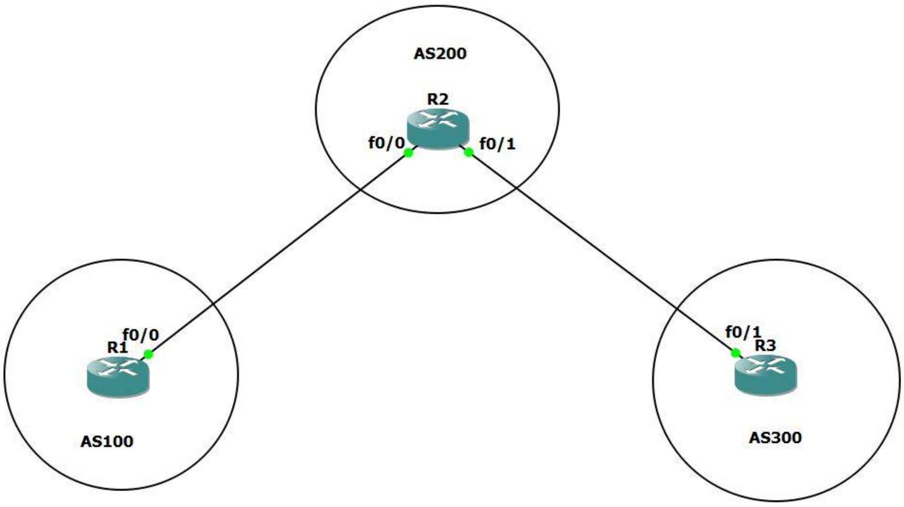
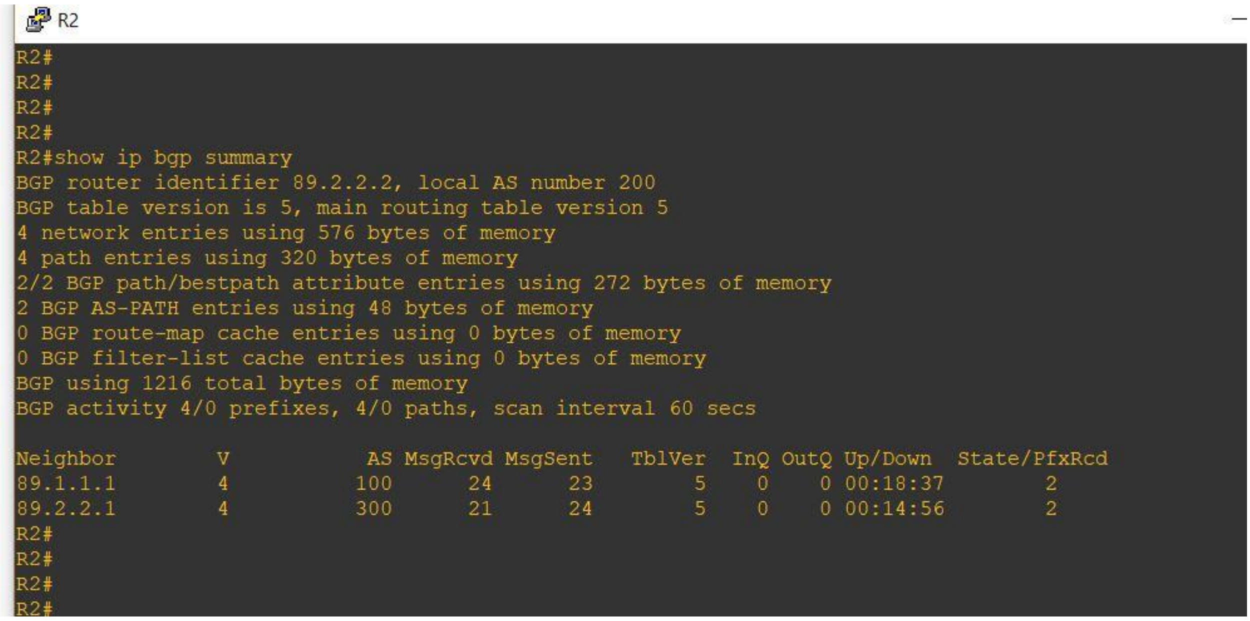


Figure 17: eBGP Peers

R1-R2-R3 belong to different ASNs forming eBGP sessions between them over TCP port: 179. Below (Image: 7), the output of the BGP neighbour table, listing the R2’s neighbours with the advertised prefixes from each of them (State/PfxRcD column).



```
R2#
R2#
R2#
R2#
R2#show ip bgp summary
BGP router identifier 89.2.2.2, local AS number 200
BGP table version is 5, main routing table version 5
4 network entries using 576 bytes of memory
4 path entries using 320 bytes of memory
2/2 BGP path/bestpath attribute entries using 272 bytes of memory
2 BGP AS-PATH entries using 48 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 1216 total bytes of memory
BGP activity 4/0 prefixes, 4/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
89.1.1.1        4          100      24      23        5    0    0 00:18:37        2
89.2.2.1        4          300      21      24        5    0    0 00:14:56        2
R2#
R2#
R2#
R2#
```

Image 6: BGP Peer Table

5.5.5 Message Types

All BGP messages have the same fixed-size header, which contains a marker field that is use for both synchronization and authentication, a length field that indicates the length of the packet, and a type field that indicates the message type. Four BGP message types have been specified in *RFC 1771* (BGP-4):

- *Open message:* a router uses this message to identify itself and specify its BGP operational parameters. Open messages are always send when the TCP session is established between neighbors.

- *Keepalive message:* if a router accepts the parameters specified in an Open message, then it responds with a Keepalive. Keepalive messages usually are exchanged at the 1/3 of the hold timer to keep sessions from expiring.

- *Update message:* is used to provide routing updates to other BGP systems, allowing routers to construct a consistent view of the network topology. Updates are sent using TCP to ensure reliable delivery. It contains unfeasible routes length, Withdraw routes, the Path Attributes and the Network Layer Reachability Information (NLRI) which is a list of IP address prefixes for the advertised routes.

- *Notification message:* it is send when an error condition is detected. Notifications are used to close an active session and to inform any connected routers of why the session is being used. Notification messages consist of the BGP header plus the error code and sub-code and data that describes the error.

5.5.6 Neighbor States

BGP forms a unique, unicast-based connection to each of its BGP-speaking peers. When BGP is configured with a neighbor IP address, it goes through a series of stages begore it reaches the desired Established state in which BGP has negotiated all the required parameters and is willing to exchange BGP routes. BGP goes through the following stages of neighbor relationship:

- *IDLE:* verifying route to neighbor. BGP refuses all incoming connections. No BGP resources are allocated in Idle state and no incoming BGP connections are allowed.

- *Connect:* BGP waits for a TCP connection to be completed. If successful, the BGP state machine moves into OpenSent state after sending the Open message to the peer. Failure in this state could result in either going into Active State or Connect, or reverting back to Idle state, depending on the failure reasons.

- *Active:* attempting connectivity to neighbor. In this state, a TCP connection is initiated to establish a BGP peer relationship. If successful, BGP sends its Open message to the peer and moves to OpenSent state. Failure can result in going to the Active to Idle states.

- *OpenSent:* open message sent to neighbor. After sending an Open message to the peer, BGP waits in this state for the Open reply. If a successful reply comes in, the BGP state moves to OpenConfirm and a keepalive is sent to the peer. Failure can result in going to the Active to Idle states.
- *OpenConfirm:* neighbor have replied with an open message. BGP waits in this state for keepalives from the peer. If successful, the state moves to Established. Otherwise, the state moves back to Idle.

- *Established:* the connection between the neighbors has been established. This is the state in which BGP can exchange information between the peers. The information can be update, keepalives or notifications.

5.5.7 Internal (iBGP) vs External (eBGP) BGP

BGP defines two types of neighbors: internal BGP (iBGP) and external (eBGP). These terms use the perspective of a single router, with the terms referring to whether a BGP neighbor is in the ASN (iBGP) or in a different ASN (eBGP). A router running BGP, behaves differently in several ways depending on whether the peer is an iBGP or eBGP. The differences include different rules about what must be true before the two routers can become neighbors, different rules about which routes the BGP best-path algorithm chooses at best and even different rules about how the routers use their attributes. The main differences between these two types of Neighborship are:

- ➢ eBGP routes have an administrative distance of 20 whereas ibgp routes are marked with an administrative distance of 200.

- ➢ eBGP peers are set with a TTL=1 and need further configuration to have a remote peer. iBGP peers are set with the maximum TTL=255 so there is no need to be directly connected.

- ➢ In iBGP a route learnt from an iBGP peer will not be advertised back to another iBGP neighbor.

- ➢ When a route is advertised to an eBGP peer by default the Next Hop is changed to local router's ip. If it is advertised to an iBGP peer then, the Next Hop remains unchanged.

5.5.8 Path Attributes

BGP Path Attributes (PA) are pieces of information that a BGP router attaches to a route to describe different prefixes included in its BGP update messages. The attributes are used to determine the best route to a destination when multiple paths exist to a particular destination. There is a variable sequence of BGP attributes in every update message except for these that carries only withdrawn routes. Each attribute is a TLV that consists of attribute type, length and value. BGP attributes have different distinctive types that defines how routes are going to use and propagate a certain attribute to its neighbors. BGP path attributes fall into the following categories [2]:

- *Well-known:* These attributes must be recognized by all the BGP implementations. Well-known PAs are separated into:
  - *Mandatory:* These attributes must be always included and carried in all BGP update messages to peers.
  - *Discretionary:* It is up to the discretion of BGP implementation to send or not these attributes in the update messages to peers.
- *Optional:* These attributes may or may not be supported by the BGP implementations. Optional Pas also fall into two sub-categories:
  - *Transitive:* BGP process has to accept the path in which it is included and should pass it on to other peers even if these attributes are not supported. That means that if any optional attribute is not recognized by a BGP implementation, then BGP looks to check if the transitive flag is set. If the transitive flag is set then BGP implementation should accept the attribute and advertise it to its other BGP Peers.
  - *Non-transitive:* If the BGP process does not recognize the attribute then it can ignore the update and not advertise the path to its peers. If the transitive flag is not set then BGP implementation can quietly ignore the attribute, it does not have to accept and advertise this attribute to its other peers.

Below is a list of path attributes that is used by every BGP router along the path to compare different network paths and select the ones to move into the BGP table and the routing table:

- *Origin*: Implies from where the route was injected into BGP (IGP, EGP or unknown). (Well-known mandatory)
- *AS_Path:* Lists ASNs through which the route has been advertised. (Well-known mandatory)
- *Next-Hop:* Lists the next-hop IP address used to reach an NLRI. (Well-known mandatory)
- *Local Preference:* a metric set and advertised throughout an AS for influencing the choice of best route for an external destination for all routers in the AS. (Well-known discretionary)
- *Atomic Aggregate:* It tags a summary NLRI as being a summary. (Well-known discretionary)

- *Aggregator:* Lists the RID and ASN of the router that created a summary NLRI. (Optional transitive)

- *Community:* Allows to share a common policy across multiple BGP peers who can be identified to be in the same group. (Optional transitive)

- *Multi-Exit Discriminator (MED):* Set and advertised by routers in one AS, impacting the BGP decision of routers in the other AS. It influences inbound policy. (Optional non-transitive)

- *Originator ID:* It is used by route reflectors to denote the RID of the iBGP neighbor that injected the NLRI into the AS. (Optional non-transitive)

- *Cluster List:* It is used by route reflection to list the route reflector cluster IDs in order to prevent loops. (Optional non-transitive)

- *Cluster ID:* The ID of the originating cluster. (Optional non-transitive)

One more attribute that is not included in this list, is the *weight attribute*. The weight is a vendor proprietary local attribute that is not propagated in BGP update messages but is used by vendor's routers for path selection.

5.5.9 Path Selection

When a BGP router learns multiple routes to the same NLRI, it must choose a single best route to reach that NLRI. BGP does not rely on a simple concept, like IGPs metric value, but rather provides a rich set of tools that can be manipulated to affect the final choice of route. The above list provides the rules that are used from BGP to determine the best path [2]:

1. Is the IP of the Next-Hop PA reachable? If not, then the route will be rejected in the decision process.

2. Prefer the path with the highest Weight PA (if it exists in the router)

3. Prefer the path with the highest Local Preference PA.

4. Prefer the route that was locally injected originated via a network or aggregate command or via redistribution from an IGP.

5. Prefer the route with the shortest AS Path.

6. Prefer the path with the lowest Origin type. IGP routes are preferred over EGP which are preferred over incomplete routes.

7. Prefer the path with lowest Multi-Exit Discriminator.

8. Prefer eBGP learned paths over iBGP paths.

9. Prefer the path with the lowest IGP metric to the BGP Next-Hop.

10. If both paths are external, prefer the oldest eBGP route.

11. Prefer the route that comes from the BGP router with the lowest RID.

12. If the originator or RID is the same for multiple paths, prefer the path with the minimum cluster list length.

13. Prefer the path that comes from the lowest neighbor address.

5.5.10 Authentication

BGP is different than the other routing protocols because you must explicitly configure the peer relationships between routers. However, it is possible to hijack an existing TCP connection (by getting the TCP sequence numbers) between two BGP peers and inject bad routes. BGP supports authentication mechanism using Message Digest 5 (MD5) algorithm. When authentication is enabled, any TCP segment exchanged between the peers is verified and accepted only if authentication is successful. The cryptographic algorithms supported in BGP are only HMAC-MD5 and HMAC-SHA1-12.

5.5.11 Router-id

Each time the BGP starts in a router, it must choose a Router-id (RID). The RID is a unique number, in x.x.x.x format, looks like an IP address, and it is used to uniquely identify an BGP process running on a router inside a routing domain. Each router in order to choose its RID follows the process:

- Checks if it is manually configured by the administrator, under routing protocol's setup.

- If not, then, the RID will be the highest IP address of the loopback interfaces.

- If there is no loopback interface, then, the RID will be the highest interface IP of the router. The interface is not mandatory to be in BGP.

- If no interface exists, it sets the router-ID to 0.0.0.0

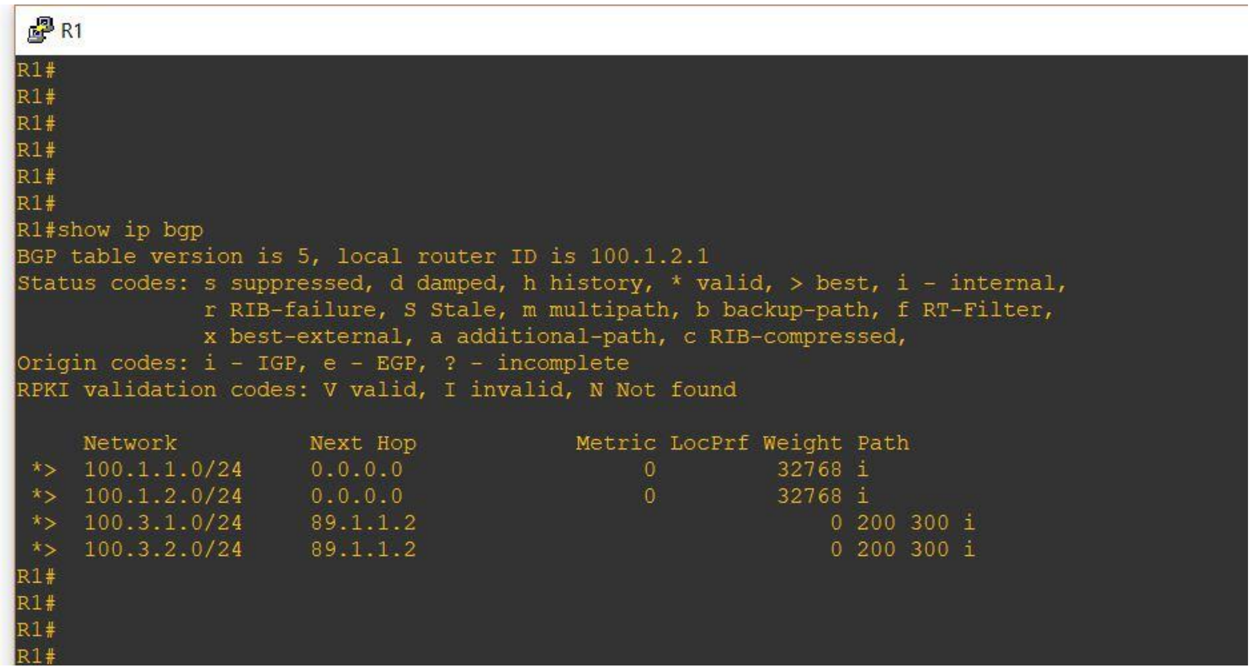
Eigrp RID is not needed in common Eigrp configuration and daily tasks. It is used when external routes are redistributed into Eigrp, RID is checked. A router cannot accept external routes from a router with the same RID.

5.5.12 Synchronization Rule

When an AS passes traffic from another AS to a third AS, BGP should not advertise a route before all routers in your AS learn about the route via an IGP. BGP waits until IGP propagates the route within the AS and then advertises it to external peers. A BGP router with synchronization enabled does not install iBGP learned routes into its routing table if it is not able to validate those routes in its IGP [7]. However, in some cases, there is no need for synchronization rule to be enabled. If there is no traffic from a different AS through own AS or all routers inside an AS run BGP, synchronization rule can be disabled. The disablement of this of this feature allows fewer routes in the IGP and helps BGP to converge more quickly.

5.5.13 BGP Table

The BGP table is the routing database of BGP. It includes all the routes learned via BGP peers and its local advertised networks. However, best paths are chosen using the default AS_Path attribute which prefers the shortest AS Path to a destination. The AS Path is shown in the right side of table, under the Path section, as shown below (Image: 6). The best routes to a destination are marked with the ***>** symbol in the first column.



```
R1#
R1#
R1#
R1#
R1#
R1#
R1#show ip bgp
BGP table version is 5, local router ID is 100.1.2.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  100.1.1.0/24     0.0.0.0                  0         32768 i
 *>  100.1.2.0/24     0.0.0.0                  0         32768 i
 *>  100.3.1.0/24     89.1.1.2                               0 200 300 i
 *>  100.3.2.0/24     89.1.1.2                               0 200 300 i
R1#
R1#
R1#
R1#
```

Image 7: BGP Table

5.5.14 BGP over IPv6 – Multi-protocol BGP (MP-BGP)

Multiprotocol BGP (or MP-BGP) was defined in *RFC 2283* and *RFC 4760*, and is an extension to BGP allowing it to advertise both IPv4 and IPv6 unicast and multicast routes. MP-BGP is also used for Mpls vpn where it uses a different address family for each «address» type. To allow these new addresses, MBGP has some new features that the old BGP doesn't have [5]:

- *Address Family Identifier (AFI)*: specifies the address family.
- *Subsequent Address Family Identifier (SAFI):* Has additional information for some address families.

- *Multiprotocol Reachable Network Layer Reachability Information (MP_UNREACH_NLRI)*: This is an attribute used to transport networks that are unreachable.
- *BGP Capabilities Advertisement*: This is used by a BGP router to announce to the other BGP router what capabilities it supports.

Since MP-BGP supports IPv4 and IPv6 we have a couple of options. MP-BGP routers can become neighbors using IPv4 addresses and exchange IPv6 prefixes or the other way around.

ANNEX I

1. Routing Protocol Redistribution

A router performs route redistribution when it uses a routing protocol information to advertise routes that were learned by some other “sources”. Other “sources” might be another routing protocol, static routes or a directly connected network. For example, a router might run both an Ospf process and a Rip process. If we configure the Ospf process to advertise routes learned by the Rip process, it is called “Ospf redistributes Rip”.

Running a single routing protocol throughout our entire network is usually more desirable that running multiple protocols both from a configuration management perspective and from a fault management perspective. However, the realities of modern networking frequently force the acceptance of multiprotocol IP routing domains. As departments, divisions, and entire companies merge, their autonomous networks must be consolidated.

In most cases, the networks that are to be consolidated were implemented differently and have evolved differently, to meet different needs or merely as the result of different design philosophies. This diversity can make the migration to a single routing protocol a complex undertaking. Route redistribution allows engineers to connect a couple of routers to the all existing routing domains, and exchange routes between them, with a minimal amount of configuration and with little disruption to the existing networks.

The main technical reason for needing redistribution is straightforward: An internetwork uses more than one routing protocol, and the routes need to be exchanged between those routing domains. The business reasons vary widely but include the following:

- Mergers with different IGPs are used.
- Mergers when the same IGP is used.
- Momentum – The enterprise has been using multiple routing protocols for a long time.
- Different company divisions are under separate control for business or political reasons.
- Connections between partners.
- Between IGPs and BGP when BGP is used to between large segments of a multinational company.
- Layer 3 Wan Mpls is implement.

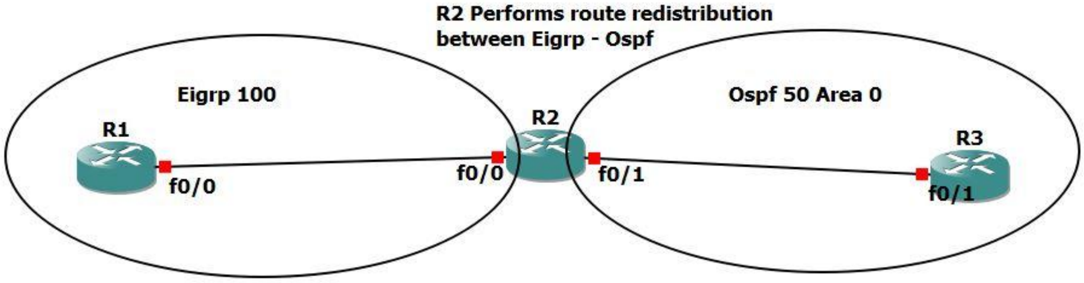


Figure 18: Route Redistribution Topology

Route redistribution requires at least one router (single point of redistribution) to provide the following:

- Use at least one working physical link with each routing domain.
- A working routing protocol configuration for each routing domain.
- Redistribution configuration for each routing protocol, mostly using a “redistribute” command, which tells the routing protocol to take the routes learned by another source of routing information and inject them into its updates.

2. Routing Update Manipulation

Routing update manipulation enables the network administrator to keep tight control over route advertisements. Any time a router is advertising or redistributing routes from one protocol, routing update manipulation tools give the power to control what routes are advertised and prevent unwanted causes or inaccurate routes to exist in a particular routing domain.

Whatever the application, route filters are a fundamental building block for creating routing policies: A set of rules that govern how packets are forwarded in a network or change the default packet forwarding behavior. Route filters work by regulating the routes that are entered into, or advertised out of, the route table. They have different effects on link-state routing protocols than they do on distance-vector routing protocols.

A router running a distance-vector protocol advertises routes based on what is in its routing table. As a result, a route filter will influence which routes the router advertises to its neighbors or receives from them. On the other hand, routers running link-state protocols determine their routes based on information in their link-state database, rather than the advertised route entries of their neighbors. Route filters have no effect on link state advertisements or the link-state database. As a result, a route filter can influence the route table of the router on which the filter is configured but has no effect on the route entries of the neighboring routers. Because of this behavior, route filters are used mostly at redistribution points into link-state domains, like an Ospf ASBR, where they can regulate which routes enter or leave the domain. Within the link-state domain, route filters have limited utility.

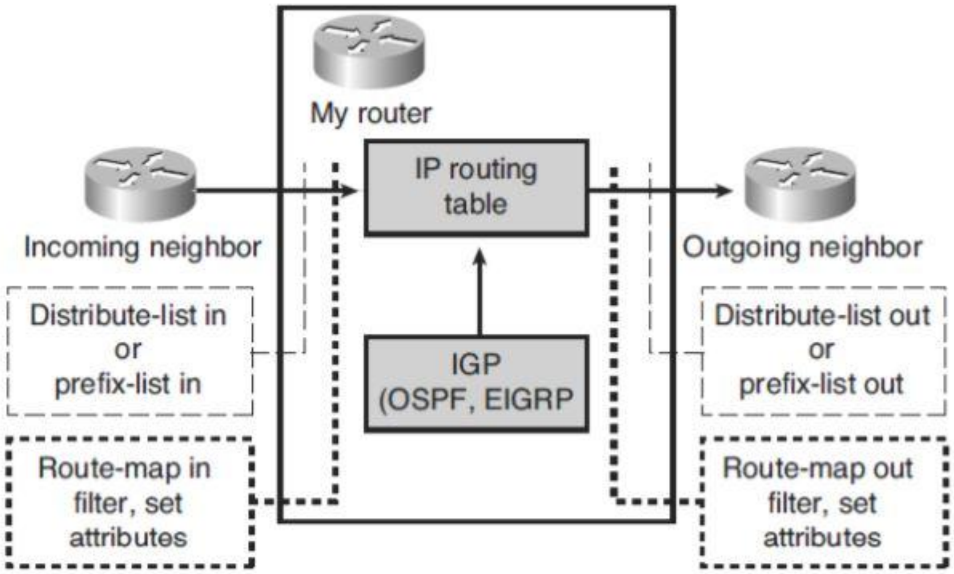

My router
IP routing table
IGP (OSPF, EIGRP
Incoming neighbor
Outgoing neighbor
Distribute-list in or prefix-list in
Distribute-list out or prefix-list out
Route-map in filter, set attributes
Route-map out filter, set attributes

Image 8: Route Filtering Process

Routing update manipulation can be accomplished by one of the following techniques:

- **Filtering specific routes**, using:

    - ***Distribute-list***: performs route filtering based on an ACL
    - ***Prefix*-list**: performs route filtering based on prefixes
    - ***Offset*-list**: increases incoming or outgoing route's metrics

- ***Route – Maps***: can be used for both redistribution and for policy routing. Similar to access-lists, they both have criteria for matching the details of certain packets and both lead to a certain action of permitting or denying. Unlike access-lists, route-maps can add to each "match" criterion, a "set" criterion that actually changes the packet or the routing information in a specified manner. Route maps allow more options for matching a given packet.

- ***Manipulating the Administrative Distance of routes***: changing the default administrative distance of advertised routes, can prevent routing loops when redistributing from a protocol with high administrative distance to a lower.

3. Policy Based Routing

Policy routes are nothing more than sophisticated static routes. Whereas static routes forward a packet to a specified next hop based on the destination address of the packet, policy routes can forward a packet to a specified next hop based on the source of the packet or other fields in the packet header. Policy routes can be linked to an extended IP access lists so that routing might be based on things as protocol types or port numbers. A policy route influences the routing decisions only of the router which it is configured.

When a packet arrives at the incoming interface of a router, the router’s data plane processing logic takes several steps to process the packet. The incoming packet actually arrives encapsulated inside a data link layer frame, so the router must check the incoming frame’s Frame Check Sequence (FCS) and discard the frame if errors occurred in transmission. If the FCS check passes, the router discards the incoming frame’s data-link header and trailer, leaving the Layer 3 packet. Finally, the router does the equivalent of comparing the destination IP address of the packet with the IP routing table, matching the longest-prefix route that matches the destination IP address.

Policy-Based Routing (PBR) overrides a router’s natural destination-based forwarding logic. PBR intercepts the packet after de-encapsulation on the incoming interface, before the router performs the FIB table lookup. PBR then chooses how to forward the packet using criteria other than the usual matching of the packet’s destination address with the FIB table. PBR chooses how to forward the packet by using matching logic defined through a route map, which in turn typically refers to an IP access control list. That same route map also defines the forwarding instructions, either the next-hop IP address or outgoing interface, for packets matched by the route map.

In the above topology (Figure:19), *Policy Based Routing* is used on R3 in order to forward HTTP traffic via ISP1 and the rest of the traffic via ISP2.

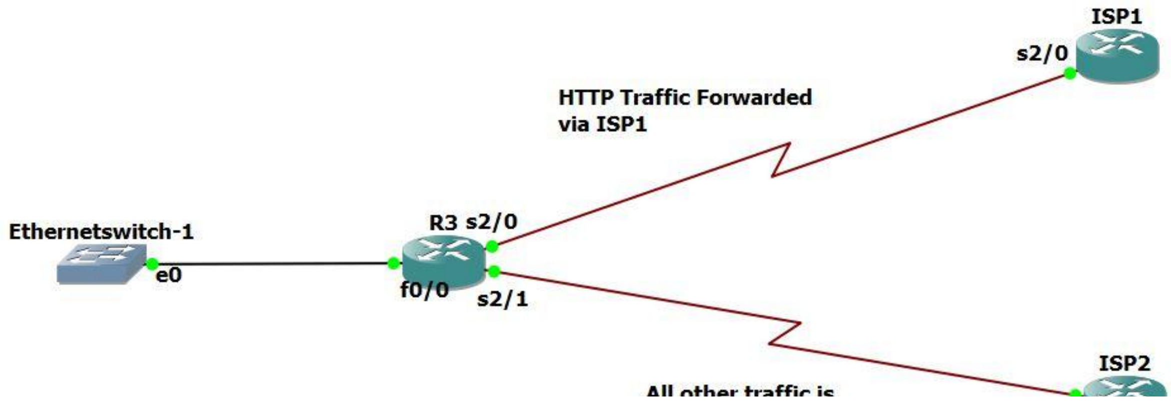


Figure 19: Policy Based Routing

REFERENCES

[1] Mark A. Dye, Rick McDonald and Antoon W.Rufi, “Network Fundamentals”, 1st Edition, 2007

[2] Jeff Doyle and Jennifer Carroll, “Routing TCP/IP”, Vol. 1, 2nd Edition, 2011

[3] Jim Kurose and Keith Ross,” Computer Networking: A Top Down Approach”, 7th Edition, 2016

[4] Wendell Odom, Rus Healy and Denise Donohue, “CCIE Routing and Switching”, 4th Edition, 2011

[5] Rick Graziani, “IPv6 Fundamentals – A Straightforward Approach to Understanding IPv6”, 1st Edition, 2012

[6] Rick Graziani and Allan Jonson, “Routing Protocols and Concepts”, 2nd Edition, 2008

[7] Kevin Wallace, “CCNP Routing and Switching, ROUTE 300-101”, 1st Edition, 2014

[8] Alvaro Retana, Don Slice and Russ White, “Advanced IP Network Design”, 1st Edition, 1999

REQUEST FOR COMMENTS

- RFC 1195 - Use of OSI IS-IS for routing in TCP/IP and dual environments, DECEMBER 1990
- RFC 1321 - The MD5 Message-Digest Algorithm, APRIL 1992
- RFC 1723 - RIP Version 2 - Carrying Additional Information, NOVEMBER 1994
- RFC 1771 - A Border Gateway Protocol 4 (BGP-4), MARCH 1995
- RFC 2328 - OSPF Version 2, APRIL 1998
- RFC 2453 - RIP Version 2, NOVEMBER 1998
- RFC 2740 - OSPF for IPv6, DECEMBER 1999
- RFC 4760 - Multiprotocol Extensions for BGP-4, JANUARY 2007
- RFC 4893 - BGP Support for Four-octet AS Number Space, MAY 2007
- RFC 5340 - OSPF for IPv6, JULY 2008
- RFC 6793 - BGP Support for Four-Octet Autonomous System (AS), DECEMBER 2012