Elphas Kipkemboi Sagas

Deploying an Open Source Router

Quagga

Helsinki Metropolia University of Applied Sciences

Bachelor of Engineering

Degree Programme In Information Technology

Thesis

04 March 2013

Author Title Number of Pages Date	Elphas Kipkemboi Sagas Deploying Open Source Router 33 pages + 2 appendices 14 February 2013
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Telecommunications and Data Networks
Instructor(s)	Erik Pätynen, Senior Lecturer

At the center of every network there is a router and a router is used to connect one network to another network. So a router has the responsibility of routing packets across different networks. The networks where this traffic is routed can either be located within the same proximity as that of the router or it can be miles away, for instance in another country.

The aim of this project was to deploy an open source router and then customize it to carry out the exact role of commercial routers, which is to route traffic between networks. Various IP routing protocols such as RIP and OSPF were tested and in order to carry out this, a small local area network comprising of two routers and two PCs were designed.

In order to experiment with this project various software and devices were needed. Some of the software needed to be downloaded and installed onto four PCs, two PCs of which were installed with an open -source routing router (Quagga) and the operating system running on the PCs was the Ubuntu Server 11.10 Server edition. The issue of compatibility between the operating system and open-source routing router was the key point as to why Quagga and Ubuntu 11.10 server edition were chosen. The same deployment can be fully implemented by small startup companies when designing their networks.

The outcome of the project showed that an open-source routing routers can be installed, configured and customized to carry out the intended purpose in a network, that is to route traffic between networks. In addition to that IP routing protocols (OSPF, RIP) worked well as expected though few problems for instance, RIPv1 worked well as compared to RIPv2 which had some shortcoming in terms of security. However the general goal of routing packets via open-source routing router (Quagga) was achieved and such project can be implemented in small companies.

Keywords	Router, Quagga, RIP, OSPF

Contents

Appendixes

1 Introduction

The functions of most companies are different but the value of a network is the same. Networks in companies are established on the basis of assisting in profit-making by minimizing input expenses and maximizing the total output, that is, the total revenue generated by a given company as compared to the capital injected into the business. With this age we are in, Information Technology (IT) has proved to be the backbone technology of every company. Most companies carry out their businesses over the Internet or within the company's internal network. With a well-established network, companies have the ability of interconnecting and routing their information or data from one point to another, therefore enhancing their abilities to transact businesses with the same or different companies located on different localities.

The objective of this project is to install, deploy and test an open-source routing platform that is freely available and can be downloaded and configured in order to achieve its full use as in the case of commercial routers. The scope of this project mainly covers the full description of what IP routing is in a network, the equipment required to establish a reliable network, some of the IP routing protocols used while routing traffic via the internet, as well as open-source routing platforms that are available to the public to download for free. To demonstrate this, a small local area network topology will be designed to mimic the real network. With this kind of network different IP routing protocols will be implemented and tested.

In this project emphasis will be given to open-source software that is free. The main target of this project is small startup companies that do not have enough capital to buy expensive routers especially when they are not sure whether their businesses will make any profit. Thus it is better to make use of the freely available software and then later upgrade to better versions. So depending on the workload of a company or simply the amount of data or information the company will be processing, different devices with different amount of processing powers will be needed, hence dictating the type of devices chosen when designing the network..

.

2 Networking

2.1 Types of Networks

2.1.1 Overview of Networks

A network in this thesis is defined as a group of two or more computer systems that are linked together by one or more communication protocols, therefore allowing them to share information or data together [1]. When connected to the Internet, packets will pass many routes and routers that are enabled and configured with different types of routing protocols. Most of these routing functionalities are normally done by a routing device commonly known as a router. A router is a device that has a role of selecting and forwarding data packets into its destination. Routers are commonly used to connect multiple networks together therefore, enabling the movement of data from one network to another, hence acting as a gateway. When a router receives an IP packet on one interface, it then determines which interface to use when forwarding the packet onto its destination. The destination can either be on the same network or a different network that can be accessed via multiple routers. [2]

When it comes to the types of networks, there are different types of networks that can be used. Some of the networks are applicable only when dealing with data within a limited geographical location, while other types of networks are ideal for sharing or sending information within a wider geographical area, as can be seen from the examples given below [2].

2.1.2 Local Area Network

Local Area Network (LAN) is a type of network which involves computers that are geographically close together and are in close proximity to each other such as in an office building, universities or small companies. A LAN is useful for sharing resources such as printers, and games as well as other applications. Expenses incurred when establishing a LAN network are less than the expenses incurred on other types of networks. A LAN operates in a limited space and is typically owned, controlled or managed by a single person or an organization. Different technologies can be used to connect LAN networks and some of them include Ethernet, a token ring among other technologies. [4]

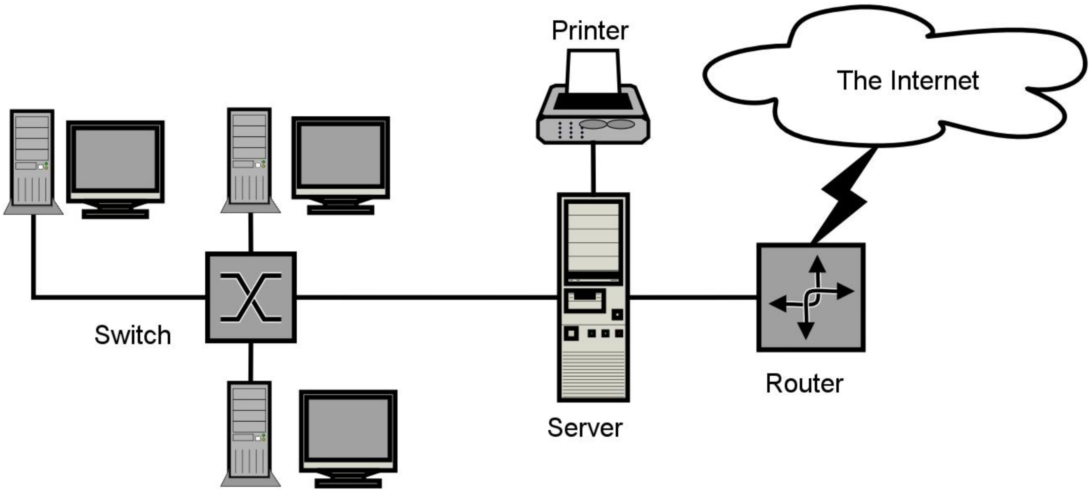


Figure 1 Local Area Network Diagram [17]

Figure 1 above shows an example of a LAN network. Whenever two or more LANs are connected together, they form another type of network known as a Wide Area Network (WAN).

2.1.3 Wide Area Network

Wide Area Network (WAN) kind of network spans a larger geographical location as compared to a LAN. A WAN network traverses public networks and commercial carriers using many different technologies [4]. A combination of several LAN networks forms WAN networks. For example Helsinki Metropolia University of Applied Sciences has many campuses in different location and each campus has its own LAN network. When these LAN networks are connected together by the use of physical layer protocols, they it spread out to form a WAN network [5].

Figure 2 below shows an example of different LANs in different locations, all connected with the use of physical layer protocols to form a WAN network. The advantages of a WAN network include ease in sharing of information and data between different local area networks since the network covers a larger area as compared to a LAN network. A WAN network provides an opportunity to the users to connect together to share devices and information, hence lowering the total costs of buying multiple devices. A disadvantage of a WAN network is the cost of setting up such a network as well as maintaining it. Because a WAN is normally a large network, constant monitoring and enhanced security is required, so to secure crucial and important information from unauthorized access. [1]

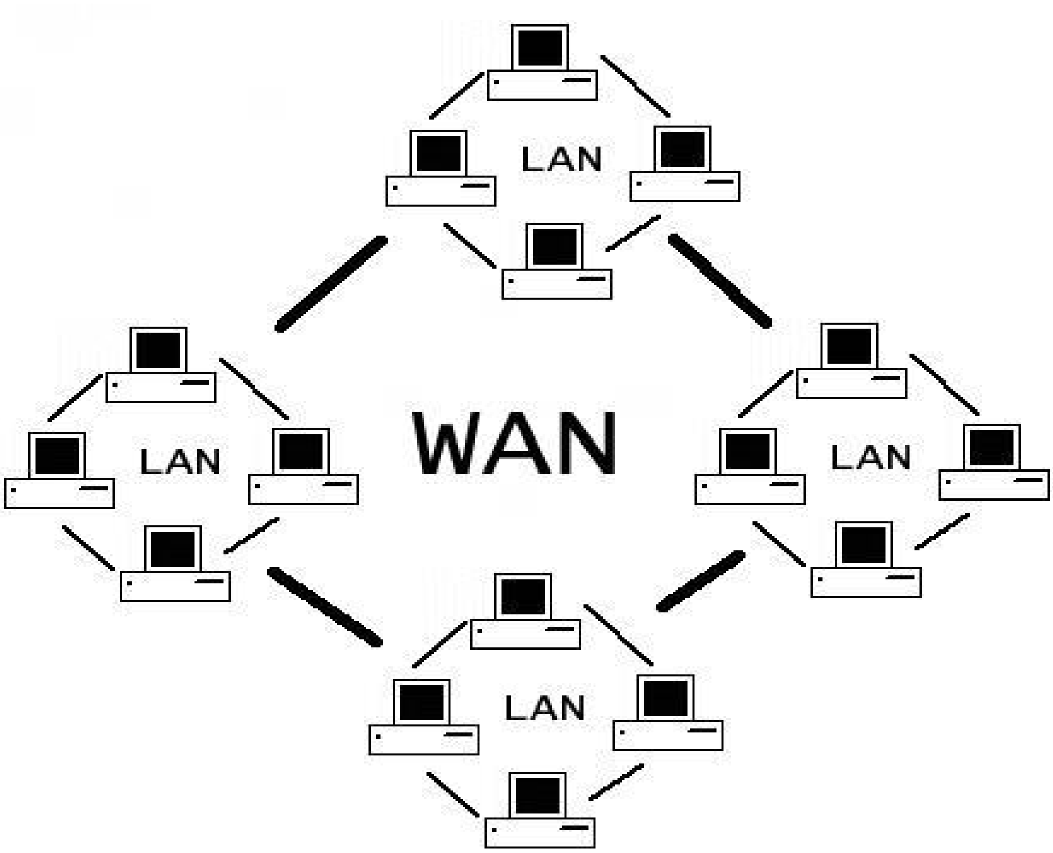


Figure 2. LAN networks connected together to form WAN. [18]

Other than the popular LAN and WAN networks there are other types of networks which include:-

- **Metropolitan Area Network** – This type of network spans a physical area larger than a LAN but smaller than a WAN; such areas include places such as cities or densely populated places. A Metropolitan area network is typically owned and operated by single entities such as a government bodies or large corporation [6].

- **Campus Area Network** – This kind of network covers multiple LAN but is smaller than a Metropolitan Area Network. A Campus area network occupies places such as university or a local business campus [6].

2.2 Network Topology

2.2.1 IP Addressing

An IP address is an identifier for computers or any network device that connects to the internet using a Transport Control Protocol /Internet Protocol (TCP/IP) Protocol. The IP address is a 32-bit numeric address expressed by four numbers separated by dots. The four numbers in an IP address are used to identify a particular network address and host in that network. IP addressing is one of the most important elements of the TCP/IP suite. With IP addresses computers running on different platforms are able to communicate together, each device in a network has a unique IP address that is used to identify that particular device as well as the network on which it is located [3, 220].

There are two standards for IP addresses: IPv4 and IPv6. IPv4 addresses contain 32 bits while IPv6 contains 128 bits. The bits consist both network identifier and host identifier. IP addresses represent network interface adapters and in the case of a router, which has at least two interfaces, each of those interfaces must have a different IP address. For instance in the topology used for an experiment which have two routers each router has two interfaces: one interface will be connected to another router's interface while the second interface will be connected to the local PC emulating LAN. Also included in the interface is Media Access Control (MAC). The MAC address is an exclusive address assigned mainly to the Network Interface Cards (NICs) by the manufacturers for identification purposes. The difference between the two is that IP addresses are inserted manually to the network interfaces through configuration scripts. An IP address can be changed at will by the administrator, while MAC is a permanent address that cannot be changed, but the two have to exist together in order for data transfer to take place. [21]

In the case of network topology on test as seen in figure 6, the topology has three networks: network 192.168.10.0/24, network 192.168.30.0/24, and network 10.2.2.0/30 and these networks are attached to other subnets. IP addresses can either be dynamic or static. The static address is the one that it is configured manually to the computer by an administrator while the dynamic routing protocol operating on a router is responsible for creating, maintaining and updating of the dynamic routing table in contrast to static routing where an administrator has the responsibility of manually inserting the routes to the routing table. [15]

2.2.2 Subnets and Sub-netting

Variable Length Subnet Masking (VLSM) is also known as sub-netting a subnet to make use of the addressing scheme. This will not only reduce the wasting of addresses but will also summarize the route and lessen the load on the Internet backbone. Subnets allow the flow of network packets based on networks that are organized into logical groups. For instance a class A network has the capability of hosting over 60 million hosts on the same network and all the hosts share the same broadcast address because they are in the same broadcast domain. However in practice it is impossible to have 60 million hosts having the same broadcast domain as most of the IP addresses will go wasted. So in order to minimize IP address wastage, a technology known as IP sub-netting was introduced, IP sub-netting enabled the administrator to make use of all the IPs by creating subnets with their own unique sub-netted network ID. Sub-netted network IDs are created by using bits from the host ID portion of the original class-based network ID. [5]

For testing purposes a small network was designed in this project. It included two PCs each running on the Linux Operating System (Ubuntu 11.10 Server Edition). These two PCs where then installed with a Quagga daemon converting them to act as routers. Each PC (Router) was connected with a computer and the two computers were used for testing the connectivity and the communication between the two routers. Both computers were running on Windows Operating systems. Figure 6 below shows the kind of network topology that was used to deploy and test Quagga as an open source routing platform.

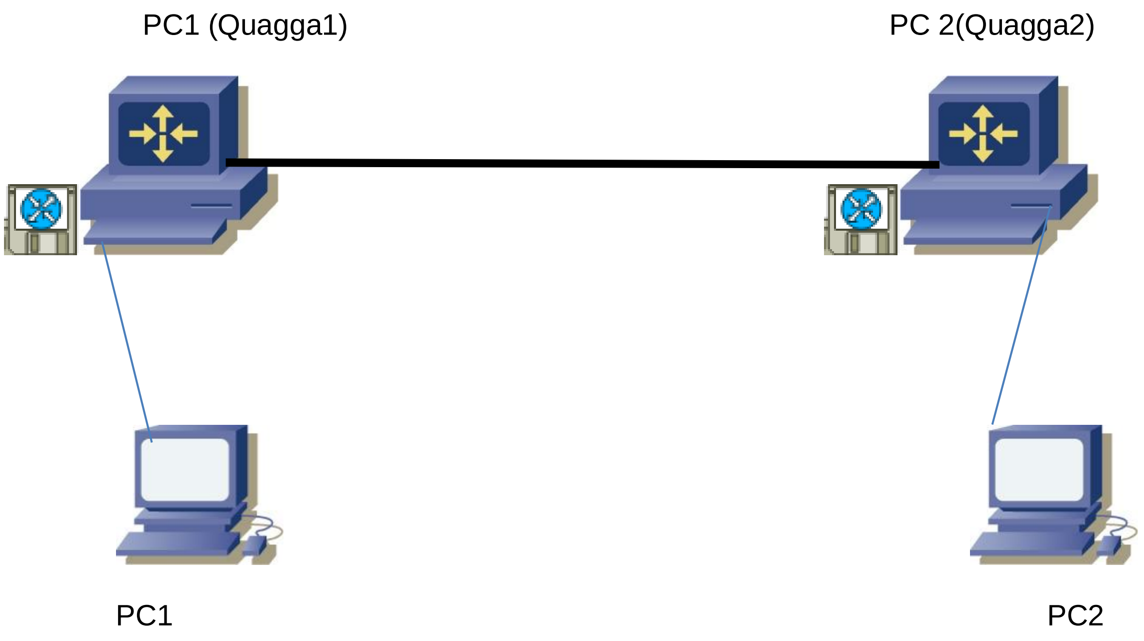


Figure 6: Target Topology used for deploying Quagga

For purposes of enabling, identifying and configuring various interfaces, IP address allocation was done. Table 1 below shows the IP addresses as they were assigned to each interface.

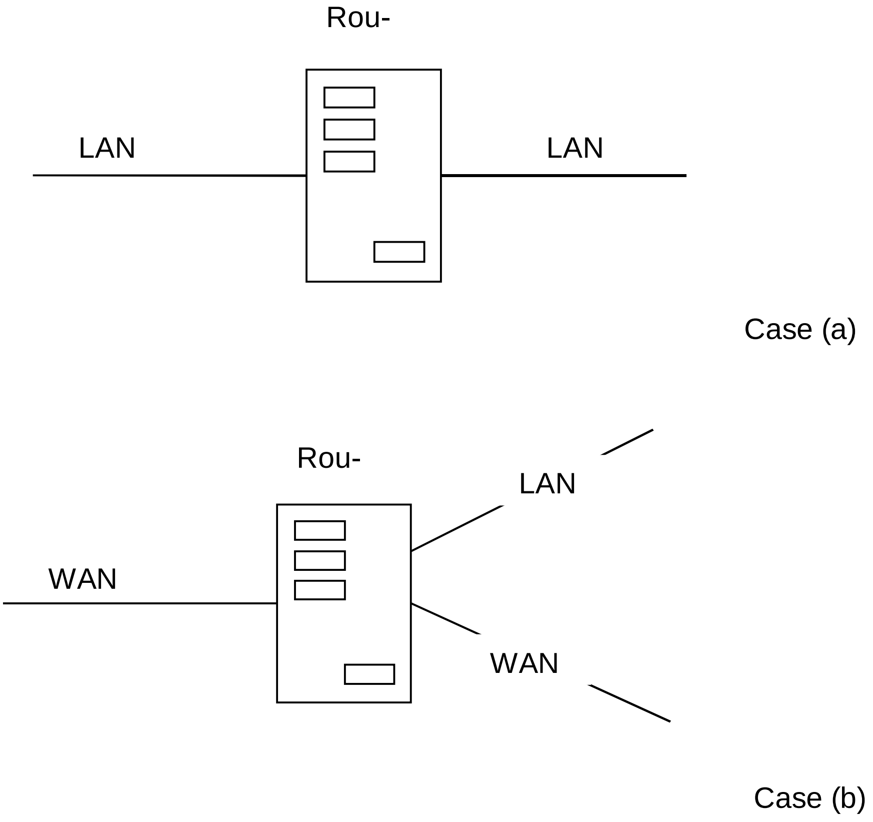
Table 1. IP routing table

Device	**Interface**	**IP address**	**Subnet Mask**	**Default Gateway**
PC1(Quagga1)	eth0	192.168.10.1	255.255.255.0	N/A
	eth1	10.2.2.1	255.255.255.252	N/A
PC2(Quagga2)	eth0	192.168.30.1	255.255.255.0	N/A
	eth1	10.2.2.2	255.255.255.252	N/A
PC1		192.168.10.10	255.255.255.0	192.168.10.1
PC2		192.168.30.10	255.255.255.0	192.168.30.1

3 IP routing

3.1 Overview of Routing

Routing is a process of finding or selecting paths in a network in which to send the information or rather traffic from one point to the other with the assistance of a device known as a router. A router is a network device that assists in the forwarding of packets from one network to another or simply to a given destination. Routers forward the traffic based on internal routing tables. To forward traffic a router must be able to read each incoming packet and decide on how best it can forward it. Normally the destination address that comes with the packet has a role of showing or directing the router on which interface that router will use to forward that particular packet [7]. Figure 3 below shows a router used to route traffic between different types of networks.



Case (a)

Case (b)

Figure 3. Types on networks that a router can be used to connect.

Figure 3 shows two routers used to route traffic between different types of networks. Case (a) shows a router used to route traffic between two LAN networks. Case (b) shows a router used to route traffic between WAN and WAN, also between WAN and LAN.

To carry out this process of routing various elements have to be considered. These elements include; IP routing protocol, which are rules that specify how routers communicate with each other, therefore allowing the routers to share the information on which route to use when routing the traffic. With this information routers are able to share routing table information, therefore constantly updating their routing table with the latest routes. Routing algorithms are used to determine the paths, and the last element is the routing database which enables storage of information that the routing algorithms have discovered. There databases normally correspond directly to the routing table entries. [7]

IP Routing protocols can be classified into two categories, namely; interior gateway protocols and exterior gateway protocol. These protocols can be further subdivided as we can see in section 3.1.2 [7]. A router sits at the edge of a network guiding traffic between two or more networks and for it to carry out those activities it must have some basic information about the network. Information about the network is normally kept in a routing table, and it includes:

- The interfaces exit interface to the destination.
- The neighboring routers for which it can learn about other existing remote networks.
- The best possible routes to each of those available remote networks.
- A procedure on how to verify and maintain any available routing information. [7;10]

Routers learn about remote networks in two ways, either dynamically using routing protocols or manually using static routes. [10]

3.1.1 Static Routing

When it comes to static routing the path between the source and the destination is predetermined and hence all related packets use the same path, unless the administrator changes it. This type of routing does not depend on the situation of the network. Once the network is configured by the administrator, it remains that way. A routing table is created, maintained and updated by the network administrator, and every static route to every network must be entered manually to the configuration files on every router for full connectivity. [8]

Static routing requires extensive planning and high management overhead. The more routers exist in a network, the more routes need to be configured, and in case a link goes down, the router does not have a mechanism to tell the packets to use another alternative and functioning link but instead sends the packets to the same nonfunctioning link, therefore causing packet drops. Static routing poses a tedious job to the network administrator since static routing is not fault-tolerant. When any changes occurs to the routing infrastructure and a link goes down, the network does not update its routing table automatically and therefore manual update by the network administrator is required to update the IP routing table.[8]

Static routes are commonly used when routing from a network to a stub network or simply a network that can be accessed via a single route. Static routing is ideal for small networks. When the network grows huge the task of updating and maintaining the routes becomes tedious. Therefore dynamic routing is preferred to static routing. [8]

3.1.2 Dynamic Routing

Dynamic routing is used to execute the same function as static routing except that in dynamic routing IP protocols assist the router to update its routing table automatically, therefore being able to recalculate a better path whenever a given link goes down. Dynamic routing protocols are usually used in larger networks to ease the administrative and operational overhead of using only static routes. This kind of routing protocol is normally configured in a router to assist in facilitating the updating and the exchange of routing information between routers. Dynamic routing protocols allow routers to dynamically share information regarding remote networks and therefore adding this information to their routing tables automatically without the need of an administrator. [8; 9]

Examples of the dynamic routing protocols include: RIP, EIGRP, and OSPF among others. Dynamic routing protocols are classified into different categories which are based on what they do and also on how they carry out routing of information. As stated previously dynamic routing protocols are divided into two major classes' namely interior gateway protocols responsible for routing information within a single autonomous system or within a single domain. The second category of a dynamic routing protocol is exterior gateway protocols that are normally used for routing information between two or many autonomous systems. [8; 10]

Depending on how these routing protocols calculate the distance between their paths, an interior gateway protocol is further subdivides into two more sub-categories namely distance vector protocol and link state protocol. Distance vector protocol routes are advertised as a vector of distance and direction. Distance in this case can be determined or rather defined in terms of a metric. For example RIPv1 and RIPv2 use a metric known as hop counts in order to determine the distance between any given router, while IGRP and EIGRP, which still fall under this category of distance vector routing protocols, use a combination of bandwidth and delay to calculate the distance between routers. [9]

IGRP is a classful distance vector internal gateway protocol that was developed in the middle of the 1980s by Cisco Systems Inc. This protocol uses a composite metric that is calculated by factoring weighted mathematical values for delay, bandwidth, load and reliability. So when the metric value is high, the route is less desirable. The IGRP newest is EIGRP acronym of Enhanced Integrated Gateway Routing protocol. [10]

Another sub-category of the interior gateway protocol is the link state protocol. Link state routing protocols build a comprehensive view of the overall network describing all possible routes alongside their costs. It uses the Shortest Path First (SPF) algorithm to create a topological database which reflects all the network routes known. So unlike distance vector routing protocols which broadcasts their information, link state protocols use multicast. When a router using link state protocols, such as OSPF and IS-IS, notices any changes in the network, it will send a multicast message notifying the concern routers about the changes that may have arisen. In this case routers do not advertise their entire routing tables but just the necessary information regarding the immediate routers. [11]

3.2 IP Routing Protocols

3.2.1 Routing Information Protocol (RIP)

Routing Information Protocol (RIP) is one of the most popular internal gateway protocols that was first developed at the University of California at Berkeley and adapted for use in the Berkeley Standard Distribution (BSD) of the Unix Operating System. Like any other routing protocol it is aimed at conveying network information to other neighboring routers. It is a dynamic and distance vector routing protocol. The RIP uses hop counts as its metric, and the number of hops is limited to 15 hops. Whenever the destination is further than 15 hops the RIP protocol assumes the destination is unreachable. [2,173].

RIP sends the complete routing update messages at regular intervals of 30 seconds and especially when the network topology changes. When a router receives routing updates it updates, its routing table to match the received information as well as reflecting the new routes. RIP routers maintain only the best route that is the route with the lowest metric value to a destination. Immediately after updating the routing table the router begins transmitting routing updates to inform other network routers of the changes. Normally these updates are sent independently of the regularly scheduled updates that RIP routers send. [8]

RIP uses a single routing metric known as hop count to measure the distance between the source and the destination. Each hop count in a path is assigned a hop count value which is normally 1. When a router receives a routing update that contains a new or changed destination network entry, the router adds one to the metric value indicated in the updates and enters the network in the routing table. RIP messages are encapsulated in a UDP segment with a source and destination port being 520. [7]

The RIP consists of three versions, RIPv1, RIPv2 and RIPng. RIPv1 and RIPv2 have some similarities, as seen above, though also they have some differences. The main difference between the two versions RIPv1 and RIPv2 is that RIPv1 uses broadcast and it is a classful routing protocol compared to RIPv2 which uses multicast and is a classless routing protocol. RIPv2 on the other hand supports authentication features unlike its predecessor RIPv1 which requires no authentication to be made. As the world of information technology transits from using IPv4 to using the latest versions of Internet protocols, that is IPv6, the RIP as a routing protocol has not remained behind, but the latest version of RIPng, which will support IPv6 protocols, has been developed. [7]

3.2.2 Open Shortest Path First (OSPF)

Open Shortest Path First (OSPF) is an IGP that uses the Link State algorithm based on the open standard designed by the Internet Engineering Task Force (IETF). The OSPF is a nonproprietary routing protocol that is commonly implemented in larger enterprise networks. OSPF is a classless routing protocol that uses the concept of area for scalability. The OSPF protocol generates and multicasts its routing updates only when a change occurs in the network topology and this happens only after every 30 minutes, compared to RIP which sends updates regularly. OSPF is based on the Short Path First (SPF) algorithm which is also referred to as Dijikstra algorithm. The SPF is used to populate the IP routing table with the best paths to each network. [19]

The major advantage of OSPF over RIP is its fast convergence and its scalability to much larger network implementations. In OSPF, when a link changes its state, the router that detects the change creates a link-state advertisement (LSA) concerning that link and sends the information to all neighboring routers using a special multicast address. Then each routing device copies the LSA and updates its link-state database (LSDB) and then forwards the LSA to all neighboring routers. [5].For effective and faster performance OSPF fragments the network into smaller groups of routers to create small areas. This will limit the traffic within the designated area, therefore not affecting the performance of the other areas**.** [19]

3.2.3 Intermediate Systems-to- Intermediate System (IS-IS)

The Intermediate System-to-System (IS-IS) routing protocols is a link state interior gateway routing protocol as opposed to a distance vector protocol such as Interior Gateway Routing Protocol (IGRP) and Routing Information Protocol (RIP). IS-IS runs the Dijikstra Shortest Path First (SPF) algorithm in order to create a database of the network's topology and to determine the best path to a specific destination in any given network. IS-IS is an ANSI ISO protocol and uses slightly different terminology as compared to the OSPF routing protocol. Some of the advantages that the link state routing protocol has over the distance vector protocol are its ability to converge faster and to avoid routing loops. This gives the link state routing protocol the ability to support large internetworks. [16]

Some of the IS-IS features include the following:

- Ability to rapidly flood its new information
- Fast convergence
- Hierarchical routing
- Highly scalability

Intermediate System-to- intermediate System Protocol is an intra-domain Open System Interconnection (OSI) dynamic routing protocol specified by the International Organization for Standardization (ISO) and the protocol is designed to operate in OSI Connectionless Network Service specified in ISO 8473.The IS-IS Routing Protocol can also be used as an IGP to support IP as well as OSI. This allows a single routing protocol to be used to support pure IP, OSI, and dual environments. Integrated IS-IS is deployed extensively in an IP-only environment in the Internet service provider (ISP) networks. The IS-IS working group of the Internet Engineering Task Force (IETF) developed the specification for Integrated IS-IS (RFC 1195). [16]

Below are steps that the IS-IS follows when updating its IP routing table:

- Routers configured to run IS-IS will send hello packets to all IS-IS enabled interfaces in order to discover neighbors and establish adjacencies.
- When the received information meets the criteria, routers sharing a common data link will become IS-IS neighbors there for forming an adjacency.
- Routers may build a link-state packet (LSP) based upon their local interfaces that are configured for IS-IS and prefixes learned from other adjacent routers.
- Generally, routers flood LSPs to all adjacent neighbors except the neighbor from which they received the same LSP. However, there are different forms of flooding and also a number of scenarios in which the flooding operation may differ.
- All routers will construct their link-state database from these LSPs.
- A shortest-path tree (SPT) is calculated by each IS, and from this SPT the routing table is built. [16]

Testing of the IS-IS was beyond the scope of the project.

3.2.4 Border gateway Protocol (BGP)

The Border gateway protocol (BGP) is one of the exterior gateway protocols, and has gone through a number of revisions, the latest version being version 4. The BGP performs inter-domain routing in Transmission-control Protocol/Internet Protocol (TCP/IP) and is considered to use a path vector routing algorithm, meaning that it tracks the path in terms of autonomous systems and not individual routers as well as advertising the path required to get to a certain destination. Its main role is to route information between multiple autonomous systems or domains, therefore exchanging routing and reachability information with other BGP systems. The BGP is a protocol used between Internet service providers and sometimes also between Internet service providers (ISP) and customer networks if needed. The BGP's routing table contains a list of known routers, the paths they can be used to reach them and a cost metric associated with the path to each router. Knowing the cost metric associated with each path will assist when choosing the best path to be used when sending packets. BGP performs three types of routing: inter-autonomous system routing, intra-autonomous system routing, and pass through autonomous system routing. [6; 19]

Inter autonomous system routing occurs between two or more BGP routers in different autonomous systems, while intra-autonomous system routing occurs between two or more BGP routers located within the same autonomous system. Finally pass through autonomous system routing occurs between two or more BGP peer routers that exchange traffic across an autonomous system that does not run the BGP. Figure 4 below shows two routers, each with a different autonomous number, and they are both connected with the BGP inter-domain routing protocol, so as to be able to exchange traffic. [14]

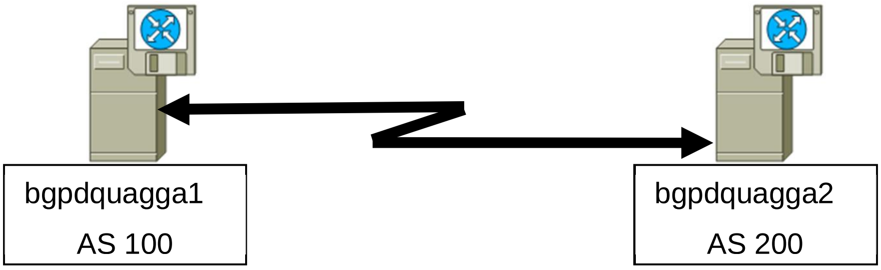


Figure 4 Autonomous systems

Figure 4 shows two routers bgpquagga1 and bgpquagga2, which are connected via a WAN point-to-point link and each router is in a different autonomous system (AS). Routers, Switches or PCs under one autonomous system are normally under one administrative control, so in this case the two routers are under different administrative control.

3.3 IP Routing suite

3.3.1 Extensible Open Router (XORP)

The eXtensible Open Router (XORP) is an open-source networking platform which stands for eXtensible Open Router Platform. The XORP is designed from the principles of software modularity and extensibility aimed at exhibiting stability. Among the operating systems that support XORP include FreeBSD, OpenBSD NetBSD, DraginFlyBSD as well as the Windows operating systems. For the purposes of traffic routing, XORP supports IPv4 and IPv6 of OSPFv2, OSPFv3, BGP4+, RIP and RIPng which is for unicast routing, and PIM-SM and IGMP/MLD for multicast. The XORP is also a fully featured platform that implements IPv4 and IPv6 routing protocols. Its primary goal is to be an open platform for networking protocol implementations and an alternative to proprietary and closed networking products. [12]

The first release of XORP was done in July 2004 under the license of General Public License (GPL) making it an open source software that can be acquired freely without any fee. XORP has a single unified command line interface (CLI) which is used for interactive configuration and operation monitoring. Its interface implements a distinct application called xorpsh, and can be invoked by multiple users simultaneously. Since XORP is an open-source routing platform, it lacks consistency in its update, therefore raising up the issue of compatibility with various operating system. The last stable and updated version was done in January 11, 2012. Though XORP appears to work on other types of operating systems, the main development platforms are FreeBSD and Linux. [12]

3.3.2 Quagga

Quagga is an open-source routing software suit that provides TCP/IP-based routing services. This routing platform supports routing protocols such as RIP v1 & v2, RIPng, OSPFv2 and v3, BGP4 and BGP4+. It also supports Intermediate System-to-Intermediate System for Unix platforms. Quagga also supports IPv4 as well as IPv6 routing protocols. When installed it acts as a dedicated router exchanging routing information with other routers using various routing protocols. This information also used to update the Kernel routing table. Some of the operating systems supporting Quagga include Unix platforms, particularly FreeBSD, Linux, and NetBSD, OpenBSD and Solaris 2.6 and higher. [2] The first release of Quagga was done in July 2010 under the General Public License version 2. Quagga consists of a core daemon known as zebra which is an abstraction layer to underlying Unix Kernel. The Zebra daemon is an IP routing manager which provides kernel routing table updates, interface lookups and redistribution of routes between different routing protocols. It works together with protocol specific routing daemons such as rip daemon, OSPF daemon and BGP daemon. [13]

4 Quagga Installation, Configuration and Customization

4.1 Quagga Daemons

When a system is installed and configured with a Quagga daemon it starts acting as a dedicated router exchanging routing information with other routers. In this project two computers running on Linux operating systems (Ubuntu 11.10 Server edition) were installed with a Quagga daemon, therefore converting the normal computers to act as routers. Different routing protocols were also put into test; some of the routing protocols tested included RIP, OSPF and BGP. Figure 5 shows a normal computer installed with a Quagga routing daemon, therefore converting it to act as a router.

Daemons are normally freeware software that is available and can be freely acquired and downloaded from the Internet. The fact that it is freeware gives a customer an opportunity to acquire it and customize it to suit his or her needs. It can be acquired with no cost unlike commercial routers.

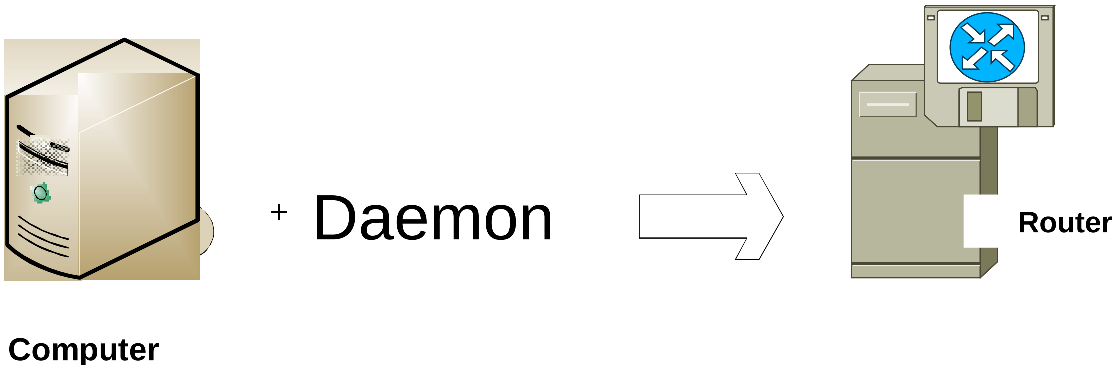


Figure 5 A normal PC installed with a Quagga daemon converting it to be a router. [13]

Normal PCs can be installed with various IP routing daemons converting them to carry out the routing process as can be seen in figure 5 above.

4.2 Quagga Architecture

Quagga architecture consists of a core routing daemon known as Zebra. The main role of these routing daemons is to initialize and dynamically maintain the kernel routing table of the network, that is, by communicating with daemons on other systems in order to exchange routing information such as the routes to a specific network or routers. Quagga is made up of collections of several daemons that work together to build a routing table. These daemons include the RIP daemons which handle the RIP protocol, the OSPF daemon which supports OSPFv2 and BGP daemon which supports the BGP-4 Protocol. For purposes of changing the kernel routing table and for redistribution of routes and the routing protocols there is a Unix Kernel routing table managed by the zebra daemon.[2; 12]

Figure 6 shows how Quagga operates. The daemons of the three dynamic routing protocols (RIP, OSPF, and BGP) that are supported by the Quagga main daemon, known as zebra, links them with the router’s kernel. Static routes are configured in the zebra configuration files and other dynamic routing protocols are done at their respective daemons. For instance to configure RIP, one must be in the RIP daemon (ripd). The same applies to OSPF and BGP. Their configurations are done in OSPF daemon and the BGP daemon respectively.

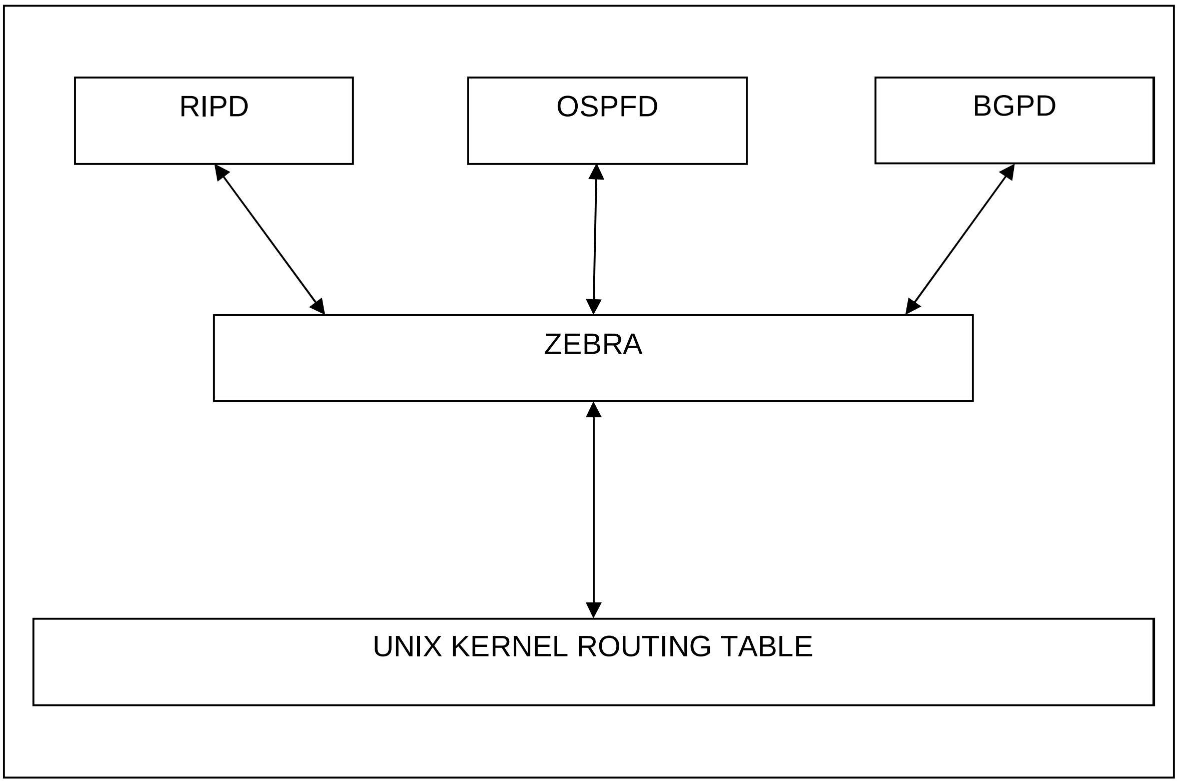


Figure 6. Quagga architecture. [13]

Before settling on Quagga as a choice to use for the experiment, a number of factors had to be considered. The first factor was the issue of software compatibility. Since the experiment involved the use of open source software, software compatibility was at the top of the list. For this case the IP routing suit software chosen had to be compatible with the Linux operating system. The second choice that dictated the choice of Quagga was its constant update by the responsible bodies, meaning that with either version of Linux compatibility would not be an issue, as noticed in the case of XORP. Lastly it needed to be open-source software which can be freely acquired without any fee. This was because the targeted customers who would use this kind of software are the startup companies who have less money to invest in their network and they need something just to let them start their business. For these reasons Quagga was the option. The official Quagga website can be located at *http://www.quagga.net*, and quagga can be downloaded and installed to a PC.

Listing 1 below shows the command at the terminal window to install Quagga from the source codes.

```
$sudo apt-get install Quagga
```

Listing 1. Quagga installation

Once downloaded and installed, Quagga files can be found in a folder named Quagga. The folder is located in `/etc/quagga`. When this command is issued `edit/etc/quagga/daemons`, the user is allowed to edit the main daemon files and therefore able to enable and disable some of the daemon files. When a *yes* is issued to a certain daemon, it means that the daemon is enabled and active, and *no* means that the daemon is inactive and for that case such a daemon cannot be used. Listing 2 below shows Zebra and RIP Daemons when active, The Zebra daemon is the main daemon in Quagga and the RIP Daemon is one of the routing protocols that Quagga can support. So for this case the OSPF and the BGP routing protocols would not be used since they were not activated.

```
zebra=yes
ospfd=no
bgpd=no
ripd=yes
```

Listing 2. Enabling routing daemons

Zebra is the main configuration file and therefore it has to be enabled all the time regardless of the IP protocol that is in use. For the daemons to work correctly, the configuration file has to be configured with the correct passwords, IP address as well as enabling and configuring the correct IP routing protocol. Listing 3 below shows the commands that can be used when copying and transferring the configuration files to the required folders.

```
$sudo cp/usr/share/doc/quagga/examples/zebra.conf.
 sample /etc/quagga/zebra.conf

 $sudo cp /usr/share/doc/quagga/examples/
 ospfd.conf.sample/etc/quagga/ospfd.conf
```

Listing 3. The commands used when locating configuration files

When the above command is used, the user is able to navigate to the folder named `example` located under `usr/share/quagga` and copy some of the examples of the configuration files located there into the folder Quagga located in a folder named `etc.` This file has to be modified and proper IP addresses inserted to the zebra.conf file this is because the main daemon (Zebra Daemon) guides the rest of the daemons. Other daemons get the IP addresses from the Zebra Daemon. For purposes of connecting via the VTY connection, passwords have to be enabled, for without passwords users cannot connect virtually to the Zebra Daemon or other available daemons such as RIP Daemon, the OSPF Daemon or the BGP Daemon. In order to allow users also to connect, users have to be given permission, since without proper permission users cannot modify or change any configurations to suit their desires. As illustrated in the commands in listing 4 a user is able to modify the permission in those given configuration files to allow the user to modify or change the configuration files. Having this permission is important when configuring the interfaces as well as enabling various IP routing daemons.

```
$ sudo chown quagga.quaggavty/etc/quagga/*.conf
$ sudo chmod 640/etc/quagga/*.conf
```

Listing 4. Giving the user full rights to modify the main configuration files

Finally the routing daemons need to be restarted, the `/etc/init.d quagga start` command is used to restart it. To verify that everything is okay `ps -fu quagga` is used. After configuring and enabling the daemons that the user intends to apply to his or her configuration, different types of IP protocols can be configured to suit the needs of the network. After fully configuring the Zebra Daemon, the final configuration scripts can be seen in appendix 1.

With the Proper VTY passwords and usernames configured to the Zebra Daemon, a user can now connect virtually to any of the daemons. Listing 5 below shows how to telnet to the Zebra Daemon. The same can be done to the RIP Daemon, the OSPF Daemon as well as the BGP Daemon.

```
root@quagga2:~# telnet localhost zebra
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Hello, this is Quagga (version 0.99.20.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

User Access Verification

Password:
Quagga2Zebra> en
Password:
Quagga2Zebra2# show run
```

Listing 5. Process of telneting to the Quagga daemons

In practice two dynamic routing protocols are put into test, both being Interior gateway protocol. The protocols include OSPF which is a link state protocol and RIP which is a distance vector protocol.

4.2.1 RIP configuration in Quagga RIPD

After making sure that the main daemon is well configured as well as up and running then other routing protocols can be configured. The procedure below shows how to configure the RIP routing protocol in the Quagga routing platform. To start with, the RIP Daemon has to be enabled or rather changed to *yes* in the main configuration files located in the Zebra Daemon. By changing the RIPD to *yes*, RIP as a routing protocol is enabled, thus giving the user ability to log in and therefore configure the RIP. When all is well, the user can log into the RIP daemon by telneting to the local host with the use of the RIPD as username and the password assigned with it.

After issuing the correct password the user is able to log into the RIP Daemon and there the RIP protocol can be configured. Listing 6 below shows how RIP was configured and in Appendix 2a and 2b the final configuration files can be seen when `show running config` is issued to ripquagga1 and ripquagga2 routers. Listing 6 shows how to configure the RIP routing protocol inclusive of network 10.2.2.0/30, network 192.168.30.0/24 for Quagga2, network 10.2.2.0/30 and network 192.168.10.0/24 for Quagga1.

```
ripdquagga1(config)#router rip
ripdquagga1(config)#network 10.2.2.0/30
ripdquagga1(config)#network 192.168.10.0/24

ripdquagga2(config)#router rip
ripdquagga2(config)#network 10.2.2.0/30
ripdquagga2(config)#network 192.168.30.0/24
```

Listing 6.RIP configuration commands

The RIP routing protocol is enabled by inserting the directly connected routes into RIP daemon.

4.2.2 OSPF Configuration in Quagga in OSPFD

The OSPF routing protocol is a link state routing protocol used for intra-domain routing; this involves routing data within a single network. The configuration of OSPF protocols on the two routers informs Quagga1 and Quagga2 that the link between the two will be used for intra-domain routing. The OSPF protocol is also responsible for sending update messages over the link informing each other about any arising changes in the network. For the case of my testing two PCs, PC1 and PC2 were installed with Quagga Daemons, thus converting the two PCs to act as routers.

For the OSPF routing protocol to work well, the OSPFD needs to acquire interface information from Zebra in order to function. Therefore Zebra Daemon must be running before the OSPFD is started up. So like the RIP, the OSPF configuration is done in OSPF Daemon. To be able to enter the OSPF daemon user will have to connect by telneting via VTY. After being allowed to access the OSPF daemon, the basic router configuration is done.

To configure OSPF a user must enter the router OSPF command when in the configuration mode then use the `network` command to enter the connected links together with their area ID, as can be seen below in listing 7. The same is done to the other connected router. Listing 7 shows how to configure the OSPF dynamic routing protocol to both quagga1 and quagga2

```
ospfdquagga1(config)# router ospf
ospfdquagga1(config-router)#network 10.2.2.0/30 area 0.0.0.0
ospfdquagga1(config-router)#network 192.168.10.0/24 area
0.0.0.0

ospfdquagga2(config)# router ospf
ospfdquagga2(config-router)# network 10.2.2.0/30 area 0.0.0.0
ospfdquagga2(config-router)#network 192.168.30.0/24 area
0.0.0.0
```

Listing 7. OSPF configuration commands

Each IP routing protocol was applied one after the other and the connectivity between PC1 and PC2 was tested using the ping command issued in both PCs as can be seen in appendix 3a and 3b.

5 Results and Conclusion

IP routing is an essential process and a requirement of any given network. When any network is designed, it is aimed at routing traffic between different networks with the help of routers as routing devices. Routers have many of the same hardware and software components as those found in a normal computer including the CPU, ROM, RAM and operating system. So in this experiment normal computers were installed with Quagga, hence giving the ability to route traffic as would have been done by dedicated commercial routers. So for any network to be termed as a reliable network, users must be able to realise the full potentiality and usability of that given network, and it must be able to route traffic effectively, faster and safely.

The goal of this thesis was to deploy and test an open-source routing platform that is freely available and can be downloaded and configured in order to achieve its full use as would have been in the case of commercial routers. Open-source software comes along with many challenges. In my case the first task was to search for and find out the available open-source routing platforms and out of a long list of available software the biggest challenge of using them was software compatibility. Therefore, before embarking on installing any open-source software, the most important point is to check whether the software is compatible with other software one intends to use with it.

The project started by downloading of Ubuntu 11.10 Server Edition to the two PCs. The two PCs were going to act as routers and few customizations were required to be done to the operating system. The step that followed was to find one open source routing platform and the first option was XORP, after a number of trials to customize and make XORP work, a number of compatibility issues came up. It was found out that XORP developers had stopped updating their source codes and therefore the software was not compatible with the latest versions of the Linux operating systems. Quagga became the chosen option because of it compatibility with the Ubuntu 11.10 Server Edition.

Finally using Quagga as a router in any given network could be an option if only the network is used to route less traffic. When more traffic is needed to be routed, bigger and more dedicated routers could be the best option. The response of Quagga when it comes to data processing seems to be slow though the IP routing protocols work well. Another issue that came up is security. Most of the latest technologies of traffic filtering and access controls do not work well in Quagga. Therefore anybody can eave through the traffic as if flows, so if a company has sensitive and private data, someone can easily compromise them. So Quagga is an ideal routing platform that can be used by small startup companies whose aim is not to secure data but to have a working network for them to exchange data.

References

1. Computer Network [Online]. Types of network; 12 July, 2011
   URL:http://www.mapsofworld.com/referrals/computers/types-of-network.html Accessed January 05, 2012

2. Bradley Mitchell (2008) Types on Network Equipments. Networking-About .com [Online]. Router, 22 September 2008.
   URL:http://compnetworking.about.com/cs/routers/g/bldef_router.htm.
   Accessed December 12, 2012

3. Carla Schroder (2007)Linux Networking Cookbook O'Reilly Media, Inc

4. Local area Network. A Webopedia Small Business IT definition [Online] 10 August 2007.
   URL:http://www.webopedia.com/TERM/L/local_area_network_LAN.html
   Accessed 5 January 2013

5. Wide Area Network Technologies. WAN protocols [Online] August 2003
   URL:http://www.ukessays.co.uk/essays/information-technology/the-wild-area network-technologies.php.
   Accessed 12 November, 2012

6. Types of Networks. SANS Technology Institute [Online] 26 October, 2007
   URL:http://www.sans.edu/research/security-laboratory/article/401-tnetwork-types.
   Accessed 12 November 2012

7. Network Technology [Online]. IP routing.
   URL:http://www.sans.edu/research/security-laboratory/article/401-tnetwork-types.
   Accessed 12 November 2012

8. TechNet [Online]. Static and Dynamic Routing
   URL:http://www.sans.edu/research/security-laboratory/article/401-tnetwork-types.
   Accessed 14 November 2012

9. Networking 101:Understanding routing [Online] 18 May 2006
   URL:http://www.enterprisenetworkingplanet.com/netsp/article.php/3607381/Networking-101-Understanding-Routing.htm
   Accessed 18 January 2013

10. Routing [Online] 26 May 2006
    URL:http://www.firewall.cx/networking-topics/routing.html
    Accessed 23 January 2013

11. Link state Routing Protocol [Online] 26 May 2006
    URL:http://www.firewall.cx/networking-topics/routing/routing-protocols/183-link-state-routing.html
    Accessed 23 January 2013

12. Getting started with XORP [Online] 11 January 2012
    URL:http://www.xorp.org/getting_started.html#getting
    Accessed 25 January 2013

13. QuaggaWiki [Online] 26 February 2012
URL:http://sourceforge.net/apps/mediawiki/quagga/index.php?title=Main_Page
Accessed 25 January 2013

14. Autonomous system [Online] 1 December 2012
URL:http://www.inetdaemon.com/tutorials/internet/ip/routing/bgp/operation/autonomous_system.shtml. Accessed 26 January 2013

15. Stephanie Crawford [1998]. What is an IP address [Online]
URL:http://computer.howstuffworks.com/internet/basics/question549.htm
Accessed 27 January 2013

16. Intermediate System-to-Intermediate System [Online]
URL:http://www.cisco.com/en/US/products/ps6632/products_ios_protocol_option_home.html
Accessed 19 January 2013

17. Stephen McQuerry [2008]. Computer Network Diagram [Online] 29 May 2008
URL:http://en.wikipedia.org/wiki/Computer_network_diagram.
Accessed 26 Jauary 2012.

18. Leroy Jones [2010]. Wide Area Network [Online] 4 June 2010
URL:http://www.technicaljones.com/2010/06/tech-term---wide-area-network
Accessed 10 November 2012

19. Cory Janssen [2010]. Dynamic Routing [Online]
URL:http://www.techopedia.com/definition/19047/dynamic-routing
Accessed 27 January 2013

20. IP routing Process step-by-step Analysis [Online] 24 May 2011
URL:http://www.firewall.cx/networking-topics/routing/181-routing-process.html
Accessed 27 January 2013

21. Carolyn Duffy Marsan [2007] IPv4 Vs IPv6 [Online] 26 Octobet 2007
http://www.networkworld.com/news/2007/102607-arguments-ipv4-ipv6.html
Accessed 28 January 2013

Appendixes

Configuration Files

Appendix 1 Zebra daemon running Configuration Files

```
Current configuration:
!
hostname Quagga2Zebra
password 8 LfkFF1rZveVW.
enable password 8 xgFr27JegUoTk
service password-encryption
!
interface eth0
 description Quagga eth0 connecting to PC2 (pc3)
 link-detect
 ip address 192.168.30.1/24
 ipv6 nd suppress-ra
!
interface eth1
 description Quagga2 eth1 connecting Quagga1 eth0
 link-detect
 ip address 10.2.2.2/30
 ipv6 nd suppress-ra
!
interface virbr0
 ipv6 nd suppress-ra
!
ip route 192.168.10.0/24 10.2.2.1/30
ip route 192.168.30.0/24 192.168.30.1/30
!
ip forwarding
!
line vty
 exec-timeout 60 0
!
end
```

Appendix 2 a RIP Daemon Running Configuration files for Quagga1

```
root@quagga1:~# telnet localhost ripd
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Hello, this is Quagga (version 0.99.20.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.


User Access Verification
```

```
Password:
ripdQuagga2> en
ripdQuagga2# show run

Current configuration:
!
hostname ripdQuagga
password 8 nKHYMBtq3i3D.
service password-encryption
!
router rip
 default-information originate
 redistribute kernel
 redistribute connected
 redistribute static
 redistribute ospf
 redistribute bgp
 network 10.2.2.0/30
 network 192.168.10.0/24
 network eth1
 network eth0
!
line vty
!
end
```

Appendix 2b RIP Daemon Running Configuration files for Quagga2

```
root@quagga2:~# telnet localhost ripd
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Hello, this is Quagga (version 0.99.20.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.


User Access Verification

Password:
ripdQuagga2> en
ripdQuagga2# show run

Current configuration:
!
hostname ripdQuagga
password 8 nKHYMBtq3i3D.
service password-encryption
!
router rip
 default-information originate
 redistribute kernel
 redistribute connected
 redistribute static
 redistribute ospf
 redistribute bgp
 network 10.2.2.0/30
 network 192.168.30.0/24
 network eth1
 network eth0
!
line vty
!
end
```

Appendix 3a showing the connectivity between PC1 and PC2 via Quagga1 and Quagga 2

```
C:\Windows\System32>ping 192.168.30.1

Pinging 192.168.30.1 with 32 bytes of data:
Reply from 192.168.30.1: bytes=32 time<1ms TTL=63
Reply from 192.168.30.1: bytes=32 time<1ms TTL=63
Reply from 192.168.30.1: bytes=32 time<1ms TTL=63
Reply from 192.168.30.1: bytes=32 time<1ms TTL=63

Ping statistics for 192.168.30.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Appendix 3b showing the connectivity between PC2 and PC1 via Quagga2 and Quagga1

```
C:\Windows\System32>ping 192.168.10.1

Pinging 192.168.10.1 with 32 bytes of data:
Reply from 192.168.10.1: bytes=32 time<1ms TTL=63
Reply from 192.168.10.1: bytes=32 time<1ms TTL=63
Reply from 192.168.10.1: bytes=32 time<1ms TTL=63
Reply from 192.168.10.1: bytes=32 time<1ms TTL=63

Ping statistics for 192.168.10.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```