CHAPTER I

1.1 Overview of Internet Challenges

The overwhelming growth of the Internet and the growing popularity of real-time applications set new challenges to the Internet community. Big Internet service providers are growing ever larger and supporting increasingly a variety of services, with different requirements both from different applications and their customers. Service providers need for a commercially viable and scalable tools to make the most of their networks in order to increase their revenues by supporting the needs of time or and mission critical applications [24]. This is because different applications have varying needs for delay, delay variation (jitters), bandwidth, and packet loss.

For example, real-time applications such as Voice over IP (VoIP) and video conferencing are extremely latency-dependent. Here, the timeliness of data delivery is an issue of utmost importance. But in the Internet, where there is no predictable traffic control, these applications do not run effectively. Service differentiation for traffic flows and performance optimization of the operational networks are very critical for the Internet to remain as successful as before. However, the Internet, particularly its core protocol IP (Internet Protocol) was never designed with Quality of Service (QoS) in mind. Instead it was originally designed as a research and educational resource, and thus the underlying technology that forms the backbone of today's Internet is largely based on that philosophy [24]. But times have changed a lot since, and service differentiation using QoS mechanisms while optimizing the operational network has become quite an important issue in the Internet for these applications to run effectively and efficiently.

On the other hand, as the Internet is required to support different types of services, effective and efficient bandwidth management tools in IP networks becomes increasingly important, especially when dealing with how to allocate the available network resources in order to optimize the overall performance of the networks. And yet, when the network has to sustain heavy traffic load, and has limited resources, the situation of having some congested links, while others remain underutilized is almost an inevitable phenomenon [24]. One of the main reasons to cause such congestion events in IP networks is that of the destination based forwarding paradigm.

In IP networks, the Interior Gateway Protocols (IGPs), such as Open Shortest Path First (OSPF), and Intermediate System-Intermediate System (IS-IS) routing protocols use destination-based forwarding algorithm, without considering other network parameters, such as the available bandwidth. In effect, all traffic between any two nodes traverses across the IGP shortest path. Hence, it is obvious that such situation can create hot spots on the shortest distance between two points, while other alternative routes may still be underutilized. As a result, degradation of throughput, and long delay, and packet losses can be noticed. In such situation, minimizing the effects of congestion by optimizing the performance of the operational networks becomes more critical.

Traffic Engineering (TE) is very important in this regard [11], and plays a key role in that it offers service providers a means for performance optimization and bandwidth provisioning. In fact, without TE, it is also difficult to support QoS on a large scale and at reasonable cost [12]. Therefore, the key to address the problem of traffic engineering is to have the ability to place the traffic onto the network as flexibly as needed, so that the congestion in the network can be minimized before it leads to poor network performance. Because minimizing congestion by optimizing the distribution of traffic on a given network is the central goal of TE [13].

In order to address the QoS issue, however, the ability to introduce connection-oriented forwarding techniques to connectionless IP networks becomes necessary. In effect, this allows IP networks to reserve resources, such as bandwidth over predetermined paths for service differentiation in order to provide QoS guarantees.

1.2 Project Background

Multiprotocol Label Switching (MPLS) is an emerging technology, which plays a key role in IP networks by delivering QoS, as well as traffic engineering features. MPLS has been developed and standardized by the Internet Engineering Task Force (IETF) to address these issues, in a more scalable and cost effective way.

A lot of research has been done on MPLS performance study, and MLPS does provide QoS and TE. In University Teknologi Malaysia, MPLS test-bed not yet been developed and used. Hence, this project is prominent since it would be a pioneer in MPLS study in our university.

1.3 Objectives of Research

Internet nowadays is facing a lot of challenges due to the gigantic numbers of users and fast growing of Internet applications. One way to overcome this problem is to provide QoS. However QoS cannot be achieved without having TE along. Due to these challenges, this project aim is to develop MPLS test-bed for network traffic engineering and to preserve QoS through explicit routing in Internet using MPLS.

1.4 Research Scopes

In order to develop MPLS test-bed, the components in MPLS domain itself need to be set up previously. For this project, four routers are needed to provide alternative routes between routers in MPLS network. Routers in MPLS test-bed which are able to forward IP packets in MPLS domain are called Label Switching Router (LSR). The routers basically based on Linux operating system since this operating system is open-source system, and the software to turn ordinary Linux PC into LSR are freely available in the Internet. Other scopes of this project are to enable the connection between routers and to create Label Switched Path (LSP). Finally IP packets passing through routers using LSP are validated using Linux command.

1.5 Thesis Outline

The contents of the project are further subdivided into chapters that described in details. Chapter I discussed about the overviews of current challenges in Internet technology. The factors why core network must provide new technology and must be improved are also specified in details. This chapter also introduced project background, objective of the research and scopes that are involved in order to finish the project. The overall project's process flow chart also shown.

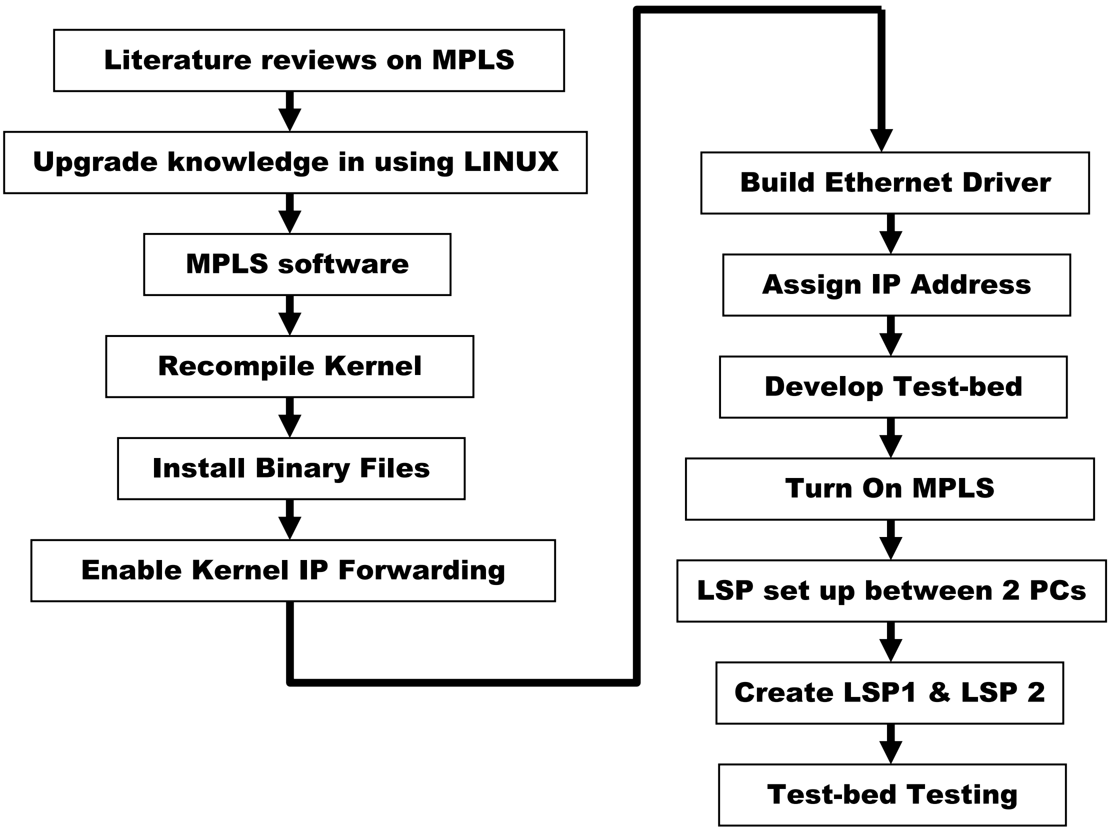
The Chapter II discussed literature studies and related works. The basic fundamentals and architecture of MPLS are explained in details. Other MPLS applications such as Virtual Private Network (VPN) and Asynchronous Transfer Mode (ATM) with IP integration also discussed. How MPLS provides Traffic Engineering (TE), Quality of service (QoS) and Class of Service (CoS) are elaborated in this chapter. Moreover, operating system used in this project is explained briefly. Finally the implementations of MPLS in Linux are explained in details such as its data structures and the processes happened when Linux machines run MPLS.

Chapter III explained about design and procedures done for this project. The steps to configure Linux machine into MPLS router are also explained. Label Switched Path (LSP) establishment between routers are described step by step for easy understanding.

In Chapter IV, results are shown and discussed. This chapter covers the results of LSP set up between two routers and the results of LSP between ingress and ingress router. Finally in this chapter, the results of Round Trip Time (RTT) for the test-bed are shown.

Lastly in Chapter V, the conclusions of this project are described. Some future works that can be done in order to enhance are explained.

1.6 Project Flow Chart



Progress of this study can be summarized into flow chart shown above. First of all, literature reviews on MPLS need to be done to acquire basic knowledge about MPLS. Journals and papers were referred to enhance the understanding and to get the overview of the project. Since this project requires Linux operating system as the platform, the knowledge using Linux must be enhanced. The understanding of Linux critical scripts that exists in Linux kernel is also very important. MPLS software is taken from Sourceforge homepage. This software package is still under development thus MPLS package version 1.946 is chosen since it is the latest up date package. Before Linux PC can be change into MPLS router, the kernel need to be compiled. The appropriate binary files are installed in order to make MPLS router to function.

Furthermore, kernel IP forwarding in Linux machine has to be change into enable state since it is disabled by default. Linux machines are connected together with appropriate IP addresses. But this can only be done when the ethernet card’s driver is installed in every Linux machine. Before LSP between two routers can be established, each router must be MPLS enabled by turning on MPLS. The connection between two routers is used again to create new LSP but now it connects ingress router to egress router.

In this study, two LSP that are LSP 1 and LSP 2 which connects ingress to egress are created. LSP 2 connects four routers while LSP 1 connects only three routers. The test-bed is tested by sending IP packet into it. When the packets captured were encapsulated with MPLS header it shown that the MPLS test-bed was well developed.