# Impact of MD5 Authentication in secured and non-secured traffic routing for the case of EIGRP, RIPv2 and OSPF routing protocols

KHALID AHMED ABU AL-SAUD

THESIS SUBMITTED IN FULFILMENT FOR THE DEGREE OF MASTER OF PHILOSOPHY

COLLEGE OF ARTS AND SCIENCE
UNIVERSITY UTARA MALAYSIA

2009

**KOLEJ SASTERA DAN SAINS**
**(COLLEGE OF ARTS AND SCIENCES)**
**UNIVERSITI UTARA MALAYSIA**

**PERAKUAN KERJA/TESIS**
**(Certification of Thesis Work)**

Kami, yang bertandatangan, memperakukan bahawa
*(We, the undersigned, certify that)*

**KHALID AHMED ABU AL-SAUD**

calon untuk Ijazah
*(candidate for the degree of)* **SARJANA SAINS (TEKNOLOGI MAKLUMAT)**

telah mengemukakan tesis/disertasinya yang bertajuk
*(has presented his/her thesis work of the following title)*

**IMPACT OF MD5 AUTHENTICATION IN SECURED AND NON-SECURED TRAFFIC ROUTING FOR THE CASE OF EIGRP, RIPV2 AND OSPF ROUTING PROTOCOLS**

seperti yang tercatat di muka surat tajuk dan kulit tesis/disertasi
*(as it appears on the title page and front cover of thesis work)*

bahawa tesis/disertasi tersebut boleh diterima dari segi bentuk serta kandungan, dan liputan bidang ilmu yang memuaskan, sebagaimana yang ditunjukkan oleh calon dalam ujian lisan yang diadakan pada : **02 Ogos 2009**

*(that the thesis/dissertation is acceptable in form and content, and that a satisfactory knowledge of the field covered by the thesis was demonstrated by the candidate through an oral examination held on*

| | | | |
|---|---|---|---|
| Pengerusi Viva *(Chairman for Viva)* | : | Dr. Fauziah Baharom | Tandatangan: *(Signature)* |
| Pemeriksa Luar *(External Examiner)* | : | Prof. Madya Dr. Hj. Mazani Hj. Manaf | Tandatangan: *(Signature)* |
| Pemeriksa Dalaman *(Internal Examiner)* | : | Encik Fazli Azzali | Tandatangan: *(Signature)* |
| Penyelia Utama *(Principal Supervisor)* | : | Prof. Madya Hatim Mohamed Tahir | Tandatangan: *(Signature)* |
| Setiausaha Panel *(Panel Secretariat)* | : | Dr. Mohd Syazwan Abdullah | Tandatangan: *(Signature)* |

Tarikh
*(Date)* : **02 OGOS 2009**

## PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the University Library may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence by the Dean of Academic, College of Arts and Sciences. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to

Dean of Academic
College of Arts and Sciences
University Utara Malaysia
06010 UUm sintok
Kedah Darul Aman

## Abstract

Routing is the process of forwarding data across an inter-network from a designated source to a final destination. Along the way from source to destination, at least one intermediate node is considered. Due to the major role that routing protocols play in computer network infrastructures, special cares have been given to routing protocols with built-in security constraints. In this thesis, we evaluate the impact of MD5 Authentication on routing traffic for the case of EIGRP, RIPv2 and OSPF routing protocols in case of secured and non-secured routing traffic. A network model of four Cisco routers has been employed and a traffic generation and analysis tools have been developed and used to generate traffic data and measure delay time, jitter and overhead. The results show that the average delay time and jitter in the secured MD5 case can become significantly larger when compared to the unsecured case even in steady state conditions. Also, the EIGRP protocol shows the minimum overhead even when the system is extremely overloaded.

## Acknowledgments

This thesis concludes my Master's Degree in Computer Network and Security submitted to the Applied Science Division, College of Arts and Science at University Utara Malaysia.

I would like to thank my supervisor Dr. Hatim Tahir for his excellent guidance, helpful advice and deeply support, my extend thanks also for the Associate Professor Dr. Suhaidi Hassan, Associate Professor and Assistant Vice Chancellor College of Arts and Sciences, at UUM.

I would like also to thank Dr. Moutaz Saleh and Dr. Mohamed Saleh for their support and enthusiasm. Dr. Moutaz has enhanced my knowledge and understanding for the art of thesis writing. Indeed, whenever I lost the sight of my thesis objectives, he proficiently led me back on the track.

My thanks to Dr. Adel El-Zoghbi for his initial support and guidance, and also, many thanks for Prof. Qutaibah Malluhi the Head of Dept. of Computer Science & Eng., College of Engineering Qatar University for his support and help.

Finally, my sincere thanks to my wife and kids, family and friends for their patience and encouragement, for them I dedicate this thesis.

## Thesis list of tables:

## Thesis list of figures

## List of Abbreviations

| | |
|---|---|
| **TCP** | Transfer Control Protocol |
| **MPP** | Markov Poisson Process |
| **HMM** | Hidden Markov Model |
| **DCE** | Data Communication Equipment |
| **DTE** | Data Terminal Equipment |
| **IGRP** | Interior Gateways Routing Protocol |
| **EIGRP** | Enhanced Interior Gateways Routing Protocol |
| **RIP** | Routing Information Protocol. |
| **RIPv2** | Routing Information Protocol version 2 |
| **OSPF** | Open Shortest Pass First |
| **MD5** | Message Digest 5 |
| **IPX** | Internetwork Packet eXchange |
| **IP** | Internet Protocol |
| **NLSP** | NetWare Link State Protocol |
| **LSA** | Link State Advertisement |
| **OSI** | Open Systems Interconnection |
| **AS** | Autonomous System |
| **DV** | Distance Vector |
| **LS** | Link State routing protocols |
| **VLSM** | Variable Length Subnet Masks |
| **IGP** | Interior Gateway Protocol |
| **EGP** | Exterior Gateway Protocol |
| **UDP** | User Datagram Protocol |
| **CIDR** | Classless Inter-Domain Routing |
| **IS-IS** | Intermediate System - Intermediate System |
| **BGP** | Border Gateway Protocol. |
| **LED** | Led Emitting Diode |
| **LAN** | Local Area Network |
| **WAN** | Wide Area Network |

| | |
|---|---|
| **QoS** | Quality of Service |
| **VPN** | Virtual Private Networks |
| **DSU/CSU** | Channel Service Unit/Data Service Unit |
| **SNMP** | Simple Network Management Protocol |
| **NLSP** | NetWare Link Services Protocol |
| **RSVP** | Resource Reservation Protocol |
| **UTP** | Unshielded Twisted-Pair |
| **STP** | Shielded Twisted-Pair |
| **WIC** | WAN Interface Card |
| **DUAL** | Diffusing Update Algorithm |
| **CPU** | Central Processing Unit |
| **BSize** | Bulk Size |
| **FP** | First Packet |
| **SP** | Step Packet |
| **MP** | Maximum packet |
| **S-RIP** | Scured-Routing Information Protocol. |

## List of publications

1. *Khalid Abu Al-Saud, Hatim Mohd Tahir, Adel Elzoghabi, Mohammad Saleh,* Performance Evaluation of Secured versus Non-Secured EIGRP Routing Protocol, in Proceedings of the 2008 **International Conference on Security & Management, SAM 2008,** Las Vegas, Nevada, USA, July 14-17, 2008. CSREA Press 2008, ISBN 1-60132-085-X.

2. *Khalid Abu Al-Saud, Hatim Mohd Tahir, Moutaz Saleh and Mohammad Saleh,* Impact of MD5 Authentication on Routing Traffic for the Case of: EIGRP, RIPv2 & OSPF, In **the Journal of Computer Sciences (JCS) 2008**, 244, 5th Avenue, Number S-207, New York, NY 10001, USA, Vol. 4(9): 721-728,.

3. *Khalid Abu Al-Saud, Hatim Mohd Tahir, Moutaz Saleh and Mohammad Saleh*, A Performance Comparison of MD5 Authenticated Routing Traffic with EIGRP, RIPv2 & OSPF, submitted in The **International Arab Journal of Information Technology (IAJIT),** accepted in January 2009 and will be publish in early 2010.

## Table of Contents

# Chapter 1

# Introduction

## 1.1 Overview

The past few years have witnessed an ever-growing reliance on computer networks for business transactions where routing plays an extensive role in these network communications. Routing is then an essential part in keeping networking infrastructures running. It is the method by which a router decides where to send a datagram. Routers are devices that direct traffic between hosts by collecting information about all the paths between a source and a destination. Based on this information, a router builds a routing table. A router may be able to send the datagram directly to the destination, if it is on one of the networks that are directly connected to the router. However, the interesting case is when the destination is not reachable directly. In this case, the router attempts to send the datagram to another router which is nearer to the destination. Thus, the goal of a routing protocol is to supply the information needed to do routing. [1], [3].

As our economy and massive infrastructure increasingly rely on the Internet, such routing protocols become of critical importance. Routing protocols, however, are difficult to efficiently secure; since an attacker attempt to inject forged routing messages into the system or may modify legitimate routing messages sent by other sources. Routing protocols are, thus, subject to threats and attacks that can harm individual users or the network operations as a whole. For instance, an attacker may attack messages that carry control information in a routing protocol to break a routers' neighboring relationship. This type of attack can impact the network routing behavior in the affected routers and likely the surrounding neighborhood as well. An attacker may also attack messages that carry data information in order to break a database exchange between two routers or to affect the database maintenance functionality where the information in the database must be authentic and authorized. Attackers can also send forged protocol packets to a router with the intent of changing or corrupting the contents of its routing table or other databases, which in turn could degrade the functionality of the router. [2], [4], [5].

To prevent such attacks, we must ensure that routers form routing protocol peering or neighboring relationships with trusted peers. One way to do this is by authenticating routing protocol messages. Enhanced Interior Gateway Routing Protocol (EIGRP), Routing Information Protocol version2 (RIPv2) and Open Shortest Pass First (OSPF) protocols support Message Digest 5 (MD5) Authentication, which uses a secret key combined with the data being protected to compute a hash. When the protocols send messages, the computed hash is transmitted with the data. The receiver uses the matching key to validate the message hash.

In fact, routing security has received varying levels of attention over the past several years and has recently begun to attract more attention specifically around the public network. Due to its dynamically changing topology, open environment and lack of centralized security infrastructure, a routing protocol is extremely vulnerable to malicious node presence and to certain types of attacks that can occur. Thus, the ongoing work on requirements for the next generation routing system and future work on the actual mechanisms for it will require well documented routing security requirements.

With the almost free flow of information and the high availability of most resources, owners and managers of enterprise networks have to understand all the possible threats to their networks. These threats take many forms, but all result in loss of privacy to a certain degree and possibly malicious destruction of information or resources that can lead to large monetary losses. A threat is then defined as a potential for violation of security, which exists when there is a circumstance, capability, action, or event that could breach security and cause harm. [6], [7].

## 1.2 Problem Statement

Routing is the operation of moving information across an inter-network from a source to a destination. Along the way, at least one intermediate node typically is encountered. Routing is often contrasted with bridging, which might seem to accomplish accurately the same thing to the casual observer. Due to the major role

that routing protocols take part in network infrastructures, special attentions have been made to routing protocols with built-in functionality to effectively secure [4]. A major concern is to avoid false routing update packets that falsely modify routing tables. Often, this is due to miss configuration or malicious purpose.

Generally speaking, to secure a routing protocol it is required that important routing information be authenticated between neighboring routers. Those various kinds of attacks actually take advantage of the lack of authenticity, integrity or confidentiality. Authentication services are primarily concerned with the providing assurances about the identity of an entity. In a routing protocol context, when a router sends out a routing message, the identity of the originator of the information should be able to be validated. Integrity services ensure that the data being transmitted is consistent with the data being received. Confidentiality service provides privacy of routing message, which uses encryption to prevent others from knowing what the routing message is.

Moreover, the current state of the ability in protecting the routing infrastructures relies on so-called best practices, which include various simplistic techniques such as passwords, Transmission Control Protocol (TCP), authentication, route filters, and private addressing to ease the most basic vulnerabilities and threats [5] [6]. Authentication occurs when two neighboring routers exchange routing information and ensures that the receiving router incorporates into its tables only the route information that the trusted sending neighbor really intends to send. It prevents a genuine router from accepting and then using unauthorized, malicious, or corrupted routing updates that may compromise the security or availability of the network. Such a compromise would lead to rerouting of traffic, or a denial of service.

In this research work, we will measure and evaluate the performance of EIGRP, RIPv2, and OSPF routing protocols in the context of: secured MD5 Authentication and non-secured situations. To meet this goal, a network test-bed model of four Cisco routers will be employed. A traffic generation and analysis tools will be developed to generate traffic data and to measure the average delay time in millisecond (ms), jitter and overhead as our performance measures of interests. Specifically, in the experiment model, a Java-based Object-oriented discrete-event

program with both client and server will be implemented at the end nodes of the network model. The network traffic, as TCP packets, will be directed from the client side to the server side which eventually calculates the major performance measures. The generation of these packets follows the Markov Poisson Process (MPP), which is a stochastic Poisson process whose rate varies according to a Markov process. The MPP can be viewed as a superposition of latent Poisson processes, which can be expressed as a non-homogeneous discretely indexed Hidden Markov Model (HMM) by partitioning time into intervals between observed events. The resultant traffic model is an exponentially distributed ON/OFF traffic where the client sends bulk traffic only during the ON periods.

## 1.3 Research Objective

The main objective of this research is to measure and evaluate the performance of routing traffic on several routing protocols for the cases of secured MD5 Authentication and non-secured. This can be carried out through studying and analyzing the available EIGRP, RIPv2 and OSPF routing protocols, first without any security rules and later with security constrains applied. Toward this main objective, the following general objectives can be obtained:

1. To investigate the effect of deploying the security MD5 authentication constrains in different routing protocols messages.
2. To specify the measure performance of metrics average delay time, jitter and overhead used in this research.
3. To develop a generic client/server program that can be employed at the end nodes of the test-bed network model for generating, monitoring and reporting the network traffic.

## 1.4 Research Scope

The scope of this research considers a test-bed network model of four Cisco 1721 routers, with Internetwork Operating System (IOS) version 12.4, which are directly connected through serial interfaces using WAN Interface Card (WIC). On each point to point connection, the clock rate on the specified Data Communication Equipment

(DCE) router terminal is set to 800,000 Hz. In addition, Client and Server, with windows OS platform, are implemented at both ends of the test-bed network model. The selected routing protocols to be configured are the EIGRP, RIPv2 and OSPF. For the authentication and security constrains we limited our research for Message Digest 5 (MD5). This is due to the fact that MD5 is a highly secured place to store the secret key since all calculations can be carried out on the routers' port avoiding traveling over a secure network.

For generating, monitoring and reporting the required TCP packet traffic, a Java-based Object-Oriented discrete event program is built. All traffic generation follows the Markovian Poisson Process (MPP) that defines a stochastic process in which network events occur continuously and independently of one another. This allows for generating network traffic bulks which has an exponentially distributed time between its arrivals during the ON periods of the source traffic model. The TCP packets' size is limited to 1000 Byte.

## 1.5 Thesis outline

The remainder of this thesis is organized into five chapters starting by introducing the literature review related to this work and ending by presenting the research conclusions and future work. The chapters are defined as the following:

In Chapter 2, we present our literature review of this research work. We start by introducing the concept of routing protocols; a protocol that specifies how routers communicate with each other to disseminate information that allows them to select routes between any two nodes on a network.

In Chapter 3, we present the research methodology we adopt for carrying out our experiment. We start by identifying the main research problems related to the routing security issues. And consequently, we defined the selected security model, MD5 Authentication. We end this chapter by identifying the selected performance evaluation technique adopted for carrying out this research, followed by illustrating the computational method for our system performance measures of interest.

In Chapter 4, we introduce in details of our test-bed network model experiment. We explain in details our experiment from both hardware and software perspectives. A Java client and server programs for generating, monitoring and reporting traffic is presented as part of this work.

In Chapter 5, we will measure and evaluate our test-bed network model. The performance measures of interest to be studied are the average delay time, jitter and overhead. Lastly, in Chapter 6, we state our research conclusions and outline future research directions.

# Chapter 2

# Literature Review

## 2.1 Introduction

A computer network is a collection of interconnected computing devices that can exchange data and share resources. In a packet-based network the computing devices communicate data by dividing the data into small blocks called packets, which are individually routed across the network from a source device to a destination device.

Routers are commonly used at interfaces between Local Area Networks (LANs) and Wide Area Networks (WANs). Internet Protocol (IP) networks are implemented with routers that interconnect physically and logically separate network segments. Routers receive data on a physical media, such as optical fiber, serial cable or Ethernet, analyze the data to determine its destination, and output the data on a physical media in accordance with the destination. In a typical packet data router, packets originating from various source locations are received via a plurality of communication interfaces [8], [9].

Each packet contains routing information, such as a destination address, which is associated with a respective communication interface of the router, e.g., by a routing table or packet forwarding protocol. In operation, the routers distinguish data packets according to network protocols and forwards traffic according to network-level addresses utilizing information that the routers exchange among themselves to find the best path between network segments. As the status of routers change in the network, the routers exchange information to reroute traffic around congested or failed routers or to route traffic to a newly activated router. A router typically includes a series of line cards in connection with a communication fabric [2].

Hence, routing is the process of selecting paths in a network along which to send data or physical traffic. Routing is actually performed for many kinds of networks, including the telephone network, the Internet, and transport networks. Routing directs logically addressed packets from their source toward their ultimate

destination through intermediary nodes; typically hardware devices called routers, bridges, gateways, firewalls, or switches. The routing process usually directs packets on the basis of routing protocols which maintain a record of the routes to various network destinations. Thus constructing routing protocols, which are held in the routers' memory, becomes very important for efficient routing. *Routing*, in a more narrow sense of the term, is often contrasted with bridging in its assumption that network addresses are structured and that similar addresses imply proximity within the network. Because structured addresses allow a single routing table entry to represent the route to a group of devices, structured addressing outperforms unstructured addressing, bridging, in large networks and has become the dominant form of addressing on the Internet, though bridging is still widely used within localized environments [3].

However, routing protocols are subject to attacks that can harm individual users or network operations as a whole. For example, an attacker may attack messages that carry control information in a routing protocol to break a neighboring e.g., peering adjacency relationship. This type of attack can impact the network routing behavior in the affected routers and likely the surrounding neighborhood. An attacker may also attack messages that carry data information to break a database exchange between two routers. Indeed, an attacker who is able to introduce bogus data can have a strong effect on the behavior of routing in the neighborhood. Another type of routing threats is called source threats that result from subverted devices; a subverted device is an authorized router that may have been broken into by an attacker. The attacker can use the subverted device to inappropriately claim authority for some network resources, or violate routing protocols, such as advertising invalid routing information [10].

Hence, securing network infrastructure is like securing possible entry points of attacks on a country by deploying appropriate defense. Computer network security is more like providing means to protect a network against outside intrusion and the civilians from getting exposed to the attacks. In fact, network security starts from authenticating any user, most likely a username and a password. Once authenticated, a stateful firewall enforces access policies such as what services are allowed to be accessed by the network users. Though effective to prevent unauthorized access, this

component fails to check potentially harmful contents such as computer worms being transmitted over the network [2].

In addition, most of the mechanisms to provide security have been available for years but have not been widely deployed or are not clearly understood probably leading to the non-deployment issue. This focuses on most of the routing protocols used in deploying IP routing architectures: EIGRP, RIPv2, and OSPF. Consequently, the lack of a common set of security requirements and methods for routing protocols has recently resulted in a wide variety of security mechanisms for individual routing protocols. Also, ongoing work on requirements for the next generation routing system and future work on the actual mechanisms for it will require well documented routing security requirements [11].

## 2.2 Routing Protocols

Routing Protocols allow routers to dynamically advertise and learn routes, determine which routes are available and which are the most efficient routes to a destination. Routing protocols provide the layer 3 network state update and populate routing tables on the layer 3 switches/routers. However, some popular layer three protocols, such as Internet Protocol (IP), Novell Internetwork Packet eXchange (IPX), and AppleTalk are called routed protocols, which transport data across the network [2].

There are two types of routing protocols: Distance Vector Routing and Link State Routing. Basically, Distance Vector protocols determine best path on how far the destination is, while Link State protocols are capable of using more sophisticated methods taking into consideration link variables, such as bandwidth, delay, reliability and load. Distance Vector protocols judge best path on how far it is. Distance can be hops or a combination of metrics calculated to represent a distance value. The IP Distance Vector routing protocols still in use today are: Routing Information Protocol (RIP v1 and v2) and Interior Gateway Routing Protocol (IGRP) [5].

On the other hand, a link-state routing is a concept used in routing of packet-switched networks in computer communications. Link-state routing works by having the routers tell every router on the network about its closest neighbors. The entire

routing table is not distributed from any router, only the part of the table containing its neighbors. Some of the link-state routing protocols are the OSPF, and IS-IS where EIGRP integrates the capabilities of link-state protocols into distance vector protocols [14]. Novell's NLSP (NetWare Link State Protocol) is also a link-state routing protocol, which only supports IPX. This type of routing protocol requires each router to maintain at least a partial map of the network. When a network link changes state, up to down or vice versa, a notification, called a Link State Advertisement (LSA) is flooded throughout the network whereby all the routers note the change, and re-compute their routes accordingly [12], [13].

Moreover, distance-vector routing protocols are simple and efficient in small networks, and require little, if any management. However, they do not scale well, and have poor convergence properties, which has led to the development of more complex but more scalable link-state routing protocols for use in large networks. Link state routing protocols, however, provide greater flexibility and sophistication than the Distance Vector routing protocols. They reduce overall broadcast traffic and make better decisions about routing by taking characteristics such as bandwidth, delay, reliability, and load into consideration, instead of basing their decisions solely on distance or hop count [11].

Accordingly, a routing protocol is a protocol that specifies how routers communicate with each other to disseminate information that allows them to select routes between any two nodes on a network. The term routing protocol may also refer more specifically to a protocol operating at Layer 3 of the Open Systems Interconnection (OSI) model which similarly disseminates topology information between routers. Typically, each router has a priori knowledge only of its immediate neighbors, and the routing protocol shares this information so that routers have knowledge of the network topology at large. For a deep discussion on the concepts of our research-related routing protocols, the following subsections are introduced [7] [12].

### 2.2.1 Interior Gateways Routing Protocol (IGRP)

The Interior Gateway Routing Protocol (IGRP) is a routing protocol that was developed in the mid-1980s by Cisco Systems, Inc. Cisco's main goal in creating

IGRP was to provide a robust protocol for routing within an Autonomous System (AS). Indeed, IGRP is kind of IGP which is a distance-vector routing protocol invented by Cisco. IGRP was created in part to overcome the limitations of RIP; maximum hop count of only 15, and a single routing metric when used within large networks. Moreover, to provide additional flexibility, IGRP permits multipath routing. Dual equal-bandwidth lines can run a single stream of traffic in round-robin fashion, with automatic switchover to the second line if one line goes down.

In addition, IGRP is considered as a Classful routing protocol. As the protocol has no field for a subnet mask, the router assumes that all interface addresses have the same subnet mask as the router itself. This contrasts with classless routing protocols that can use variable length subnet masks. Classful protocols have become less popular as they are wasteful of IP address space. Consequently, Cisco developed Enhanced IGRP in the early 1990s to improve the efficiency of IGRP. Figure 2.1 illustrates the IGRP protocol structure.

| 8 bits | 16 bits | 24 bits | 32 bits |
|---|---|---|---|
| Version | Opcode | Edition | ASystem |
| Ninterior | Nsystem | Nexterior | Checksum |

**Figure 2.1: IGRP Protocol Structure**

As figure 2.1 depicted, the IGRP protocol structure contains the following fields:

- Version - IGRP version number (currently 1).
- Opcode - Operation code indicating the message type: 1 Update; 2 Request.
- Edition - Serial number which is incremented whenever there is a routing table change.
- ASystem - Autonomous system number. A gateway can participate in more than one autonomous system where each system runs its own IGRP. For each autonomous system, there are completely separate routing tables.

- Ninterior, Nsystem, Nexterior - Indicate the number of entries in each of these three sections of update messages. The first entries (Ninterior) are taken to be interior, the next entries (Nsystem) as being system, and the final entries (Nexterior) as exterior.
- Checksum - IP checksum which is computed using the same checksum algorithm as a UDP checksum [14], [15].

### 2.2.2 Enhanced Interior Gateways Routing Protocol (EIGRP)

The EIGRP using Diffusing Update Algorithm (DUAL) algorithm, referred to as an advanced Distance Vector (DV) protocol, offers radical improvements over IGRP. Traditional DV protocols such as RIP and IGRP exchange periodic routing updates with all their neighbors, saving the best distance or metric and the vector or next hop for each destination. EIGRP differs in that it saves not only the best least-cost route but all routes, allowing convergence to be much quicker. Further, EIGRP updates are sent only upon a network topology change; updates are not periodic. Instead, EIGRP sends partial updates only when the metric for a route changes. Propagation of partial updates is automatically bounded so that only those routers that need the information are updated. As a result of these capabilities, EIGRP consumes significantly less bandwidth than IGRP. Figure 2.2 illustrates the EIGRP protocol structure.

| **8 bits** | **16 bits** | **32 bits** |
|---|---|---|
| Version | Opcode | Checksum |
| Flags | | |
| Sequence number | | |
| Acknowledge number | | |
| Asystem: Autonomous system number | | |
| Type | | Length |

**Figure 2.2: EIGRP Protocol Structure**

As figure 2.2 depicted, the EIGRP protocol structure contains the following fields:

- Version - The version of EIGRP.
- Opcode - indicating message type: 1 Update. 2 Reserved. 3 Query. 4 Hello. 5 IPX-SAP.
- checksum - is computed using the same checksum algorithm as a UDP checksum
- Flag - Initialization bit and is used in establishing a new neighbor relationship
- Sequence number - used to send messages reliably
- Acknowledge number - used to send messages reliably
- Asystem - Autonomous system number. A gateway can participate in more than one autonomous system where each system runs its own IGRP.
- Type - Value in the type field: 1 EIGRP Parameters. 2 Reserved. 3 Sequences. 4 Software version 5 Next Multicast sequence.
- Length - Length of the frame.

Also, EIGRP updates carry subnet mask information which allows EIGRP to summarize routes on arbitrary bit boundaries, support classless route lookups, and allow the support of Variable Length Subnet Masks (VLSM). EIGRP use five metrics when performing path forwarding. These EIGRP metrics are shown in Table 2.1.

Table 2.1: EIGRP Metrics

| **Metric** | **Value** |
|---|---|
| Bandwidth | In units of kilobits per second; 10000 for Ethernet |
| Delay | In units of tens of microseconds; for Ethernet it is100x10 microseconds=1 ms |
| Reliability | 255 for 100 percent reliability |
| Load | Effective load on the link expressed as a number from 0 to 255 (255 is 100 percent loading) |
| MTU | Minimum MTU of the path; usually equals that for the Ethernet interface, which is 1500 bytes |

[16] [17]

Since its development, EIGRP is known to converge as quickly as a link-state protocol in a medium-scale network while maintaining loop freedom at every instant. The protocol, which is deployed in part of Cisco Systems engineering network, based on three main elements: a transport algorithm that supports the reliable exchange of messages among routers, the diffusing update algorithm, which computes shortest paths distributedly, and modules that permits the operation of the new routing protocol in a multiprotocol environment. Furthermore, EIGRP provides multiple paths to every destination that may have different weights, and thus many sites have run EIGRP in both real-world internetwork and laboratory environment [18].

For instance, to evaluate EIGRP performance under a very dynamic network, a simulation model has been employed in [19]. The simulated network was a composite of wired and wireless networks, and the results hold for both types of media. The study shows that the host mobility using route updates is a feasible method to achieve seamless mobility and continuous connectivity for users of mobile wireless devices as they move within an AS. Moreover, the EIGRP overhead incurred from mobility is minimal as all of EIGRP query and reply messages are small. Consequently, the research results in [19] showed that EIGRP converges faster than a single TCP timeout in most cases.

In the year of 2000, network architects state that EIGRP is among routing protocols which are implemented in approximately half of the networks [20]. They claimed that EIGRP is not only an enterprise-oriented routing protocol, but also is protocol that can be used in service provider environments because it has fewer topology limitations than others.

Recently, a number of research works has been done on analyzing the EIGRP performance. For instance, M. Gouda et al in [21] presented a simple theory and several applications for maximizable routing metric. They showed that the composite metric used by IGRP and EIGRP is not maximizable. Analyzing the Diffusing Update Algorithm (DUAL) for using nonmonotonic composite routing metrics, EIGRP may not behave as expected. Their proofs of the necessity of monotonicity was based upon the definition of maximizable, which requires that there exist maximum metric trees with respect to all edge weight assignments to all networks.

Nigel Houlden et al [22], consider the use of compound cost functions in routing calculations and develop the theoretical principals of optimal end-to-end EIGRP routing protocol. To determine the path cost function of EIGRP, the formula is generally stated as:

$$C = \left( k_1 b + \frac{k_2 b}{256 - l} + k_3 d \right) \frac{k_3}{r - k_4}$$

where b is the minimum bandwidth measured in kilobits per second; l the load on the link expressed as a number from 0 to 255 (255 is 100 percent loading), d the total delay in unit of tens of milliseconds, and r the reliability along the length of the path 255 for 100 percent. k1, k2, k3, k4 and k5 are administrator-configurable coefficients (although the values must be consistent across the domain). However, even this calculation is complicated by the need to scale bandwidth and delay as b = (256 x 108) / b0 and d = 256d0, where b0 and d0 are the measured or configured values; the 256 arises from a storage difference (from IGRP to EIGRP) between 24 and 32 bits. Indeed, it is claimed that the default coefficient values of k1=1, k2=0, k3=1, k4=0 & k5=0 lead to the simplified path cost of C = b + d.

Eventually, the research work in [23] developed a complete model and associated tools for characterizing interconnections between routing instances based on analysis of router configuration data. The experimental results emphasize the urgent need for more research to improve this model safety and flexibility to support important design objectives. The work discovers that the high vulnerability of route redistribution to routing abnormalities has resulted in complex configurations. Furthermore, some of the complex configurations can still be vulnerable to routing instabilities. These empirical results end to strongly suggest making EIGRP routing protocol safe and robust.

### 2.2.3 Routing Information Protocol (RIP)

RIP is a relatively old, but still commonly used, Interior Gateway Protocol (IGP) created for use in small and homogeneous networks. It is a classical Distance-Vector routing protocol. RIP uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. Since RIP uses UDP as its delivery mechanism, the

routing updates sent to the neighboring routers are not guaranteed. The sending of the RIP table entries between routers defaults to 30 seconds after the initial startup of the router. This advertising of routes occurs also between two routers when a router becomes active on a connection to an already active router. RIP sends the updates to the interfaces in the specified networks. If an interface's network is not specified, it will not be advertised in any RIP update [14] [15].

Due to the performance limitations of the traditional RIP protocol, the new RIP version2 protocol was introduced. RIP version 2 derives from RIP, which is an extension of the Routing Information Protocol (RIP) intended to expand the amount of useful information carried in the RIP messages and to add a measure of security. The RIP Version 2 supports Key management, plain text and Message Digest (MD5) authentication, route summarization, Classless Inter-Domain Routing (CIDR), and variable-length subnet masks (VLSMs). Figure 2.3 illustrates the RIP protocol structure [10] [11].

| **8 bits** | **16 bits** | **32 bits** |
|---|---|---|
| Command | Version | Unused |
| Address family identifier | | Route tag (only for RIP2; 0 for RIP) |
| IP address | | |
| Subnet mask (only for RIP2; 0 for RIP) | | |
| Next hop (only for RIP2; 0 for RIP) | | |
| Metric | | |

**Figure 2.3: RIP Protocol Structure**

As figure 2.3 depicted, the RIP protocol structure contains the following fields:

- Command - The command field is used to specify the purpose of the datagram. There are five commands: Request, Response, Trace on, Trace off and Reserved.
- Version - The RIP version number. The current version is 2.

- Address family identifier - Indicates what type of address is specified in this particular entry. This is used because RIPv2 may carry routing information for several different protocols. The address family identifier for IP is 2.
- Route tag - Attribute assigned to a route which must be preserved and re-advertised with a route. The route tag provides a method of separating internal RIP routes, routes for networks within the RIP routing domain, from external RIP routes, which may have been imported from an EGP or another IGP.
- IP address - The destination IP address.
- Subnet mask - Value applied to the IP address to yield the non-host portion of the address. If zero, then no subnet mask has been included for this entry.
- Next hop - Immediate next hop IP address to which packets to the destination specified by this route entry should be forwarded.
- Metric - Represents the total cost of getting a datagram from the host to that destination. This metric is the sum of the costs associated with the networks that would be traversed in getting to the destination. The RIP metric is composed of hop count, and the maximum valid metric is 15. Anything above 15 is considered infinite; we can use 16 to describe an infinite metric in RIP [15], [16], [17].

As being the most popular distance vector routing protocol, RIP gets a deep concern from many researchers since its creation. For instance, the research work in [25] proved that such existing protocols are insecure due to the lack of strong authentication and authorization mechanisms and the difficulty, if not impossibility, of validating routing messages which are aggregated results of other routers. Consequently, the researchers introduced a secure routing protocol, namely S-RIP, based on a distance vector approach. In S-RIP, a router confirms the consistency of an advertised route with those nodes that have propagated that route. The threat analysis and simulation results showed that in S-RIP, a well-behaved node can uncover inconsistent routing information in a network with many misbehaving nodes assuming no two of them are in collusions, with relatively low extra routing overhead.

In addition, the research work in [26] developed a simple and effective approach called RIP with Triangle theorem checking and Probing RIP-TP to detect and identify suspicious or invalid new routing messages in RIP routing protocol and use

probing messages to verify the correctness of the messages. They evaluated the effectiveness of RIP-TP through simulation using various faulty node behaviors, link failure dynamics and network sizes. Their design emphasizes effectiveness, simplicity, low overhead, backward compatibility with the standard RIP protocol, and supports for incremental deployment. The results showed that, in the worst case, RIP-TP can effectively detect 95% or more invalid routing announcements. As they demonstrated, existing RIP routing protocols can be enhanced with effective fault detection capability. However, a year later, some researchers argue that distance vector protocols are poor candidates for detecting faults because a router has no way to verify the validity of the distance information [27].

Recently, Abdelaziz Babakhouya et al [28] proposed a new approach called S-DV to Secure Distance Vector Routing Protocols which provides both protections from internal and external attackers. This mechanism is less expensive that offers a deterministic detection of distance fraud than the approach proposed in S-RIP [26]. Moreover, SDV routers use a new metric that they designate by Security Indicator, to prefer a choice of a secure route than a shortest one which has been subject to frequent attacks. This reduces the overhead compared to S-RIP and increases the scalability of RIP protocol. This is guaranteed through metric which measures the frequency of malicious routing updates, received from each neighboring node.

### 2.2.4 Open Shortest Pass First (OSPF)

The OSPF is an interior gateway protocol used for routing between routers belonging to a single Autonomous System. OSPF uses link-state technology in which routers send each other information about the direct connections and links which they have to other routers. Each OSPF router maintains an identical database describing the Autonomous Systems topology. From this database, a routing table is calculated by constructing a shortest path tree. OSPF recalculates routes quickly in the face of topological changes, utilizing a minimum of routing protocol traffic. OSPF provides support for equal-cost multi-path. An area routing capability is also provided, enabling an additional level of routing protection and a reduction in routing protocol traffic. In addition, all OSPF routing protocol exchanges are authenticated and the OSPF metric is a cost value based on $10^8$/bandwidth of the link in bits/sec.

OSPF allows sets of networks to be grouped together. Such a grouping is called an area. The topology of an area is hidden from the rest of the Autonomous System. This information hiding enables a significant reduction in routing traffic. Also, routing within the area is determined only by the area's own topology, lending the area protection from bad routing data. Figure 2.4 illustrates the OSPF protocol structure.

| **8 bits** | **16 bits** | **24 bits** |
|---|---|---|
| Version No. | Packet Type | Packet length |
| Router ID | | |
| Area ID | | |
| Checksum | | AuType |
| Authentication (64 bits) | | |

**Figure 2.4: OSPF Protocol Structure**

As Figure 2.4 depicted, the OSPF protocol structure contains the following fields:

- Version number - Protocol version number (currently 2).
- Packet type - Valid types are as follows: 1 Hello, 2 Database Description, 3 Link State Request, 4 Link State Update, 5 Link State Acknowledgment.
- Packet length - The length of the protocol packet in bytes. This length includes the standard OSPF header.
- Router ID - The router ID of the packets source. In OSPF, the source and destination of a routing protocol packet are the two ends of an (potential) adjacency.
- Area ID - identifying the area that this packet belongs to. All OSPF packets are associated with a single area. Most travel a single hop only.
- Checksum - The standard IP checksum of the entire contents of the packet, starting with the OSPF packet header but excluding the 64-bit authentication field.

- AuType - Identifies the authentication scheme to be used for the packet.
- Authentication - A 64-bit field for use by the authentication scheme.

[15], [16], [17].

OSPF is a link state routing protocol which received particular attention in the literature. For instance, the authors in [29] present a methodology and examination of how OSPF routing protocol is used in operational networks. They demonstrate and present a methodology for working with the configuration files of production networks. The results showed that the conventional model of interior gateway protocols is insufficient to describe the diverse set of mechanisms used by architects, and argued that it opens paths towards new understandings of network behavior and design. In [30], the study analyzed and investigates for OSPF routing protocol from both stability and dynamics view. The analysis was based on large-scale simulations of OSPF, and careful design of experiments to perform an efficient search for the best parameter settings of routing protocol. The results defined the number of routing updates as the metric to minimize in the search for the best parameter settings and found that the link status changes propagated heavily from the OSPF effects of link weight changes

The research work in [31] also describe an experimental evaluation of using an approximation to OSPF Multi-Topology Routing (MTR), to provide more load balancing choices within the OSPF framework and a policy-based route map prototype of MTR-like routing, to better distribute traffic load in congested networks. MTR works by overlaying multiple logical OSPF topologies on a single physical topology and by consistently mapping packets based on header bits to a logical topology at each hop. Consequently, explore the performance benefits and scaling trends of MTR traffic management in a small scale network setting. The results have shown that MTR optimization provides moderate to significant performance improvements under most test conditions.

Moreover, the research work in [32] present an approach to improve the realism of the on–demand methods, without incurring the memory overhead of the routing protocol method, when forwarding of packets from a source to a destination in the simulated topology firstly; by implementing a model of OSPF routing protocols,

secondly; based on revised global topology knowledge within the simulator to simply re-compute routing information when the topology changes occur, and lastly; use on–demand routing computations using global topology knowledge. Both of the first two suffer from excessive memory requirements for routing table storage, which can be in the extreme case. The last two suffer from unrealistic routing decisions in the presence of topology changes, due to the use of instantaneous global topology knowledge. The proposed design can be tuned to model time delays needed for routing protocols to converge on new routing information. The effectiveness in terms of dropped packets due to incomplete information and in terms of memory and CPU overhead was demonstrated. The results showed that the overhead is minimal, both in terms of memory usage and CPU overhead.

Recently, the researchers in [33] built an experimental network to implement various routing activities which are monitored and analyzed via logging and snmp trap analysis systems to examine the performance and security issues of various existing routing protocols including RIP, OSPF, and EIGRP. Various routing performance metrics are evaluated and analyzed via the logged and trapped information. Two kinds of problems will occur in the routing security issue. One is to increase the overhead of routing maintenance in order to degrade or even disable the routing service, and the other is to modify the routing information such that packets will be forwarded incorrectly or dropped. They build two simple attacks to evaluate and analyze the overhead for existing routing protocols: RIP, OSPF and EIGRP. Eventually, they proposed a log-based analysis system in which routers' routing logs can be used to monitor the activities of routing protocols in order to detect malicious and abnormal routing behaviors. They conclude that the routing protocols authentication is required to protect legitimate routing packets and reduce the probabilities of being attacked.

### 2.2.5 Comparison of Routing Protocols

Table 2.2: illustrates a summary of the above routing protocols with respect to the most common interior routing protocols criteria's.

Table 2.2 Comparison of Common Routing Protocol Features

| Criteria | IGRP | EIGRP | RIPv2 | OSPF |
|---|---|---|---|---|
| Type | DV* | DV | DV | LS** |
| Convergence Time | slow | fast | Slow | Fast |
| VLSM*** | no | Yes | Yes | Yes |
| Bandwidth Consumption | high | low | High | Low |
| Resource Consumption | low | low | Low | High |
| Multi-path Support | yes | yes | No | Yes |
| Scales Well | yes | yes | No | Yes |
| Proprietary (Cisco prod.) | yes | yes | No | No |
| Routers Non-IP Protocols | No | yes | No | No |

* DV = Distance Victor routing protocols [3]

** LS = Link State routing protocols.

*** VLSM = Very Large Subnet Mask.

## 2.3 Routing Protocol Authentication

Most routing protocols incorporate neighbor authentication to protect the integrity of the routing domain. Authentication occurs when two neighboring routers exchange routing information and ensures that the receiving router incorporates into its tables only the route information that the trusted sending neighbor really intends to send. Authentication prevents a legitimate router from accepting and then using unauthorized, malicious, or corrupted routing messages that may compromise the security or availability of the network; for example, having an unauthorized device send routing information that makes the legitimate router believe that the best route for certain traffic is via an alternative path that may or may not exist. Such a compromise would lead to rerouting of traffic, a denial of service, or just giving access to certain packets of data to an unauthorized person.

When neighbor authentication is configured, the router authenticates the source of each routing update packet it receives. This is accomplished by the exchange of an authentication key, sometimes referred to as a shared secret, which is known to both the sending and the receiving routers. Two types of neighbor authentication are typically used: plaintext authentication and cryptographic authentication typically using the Message Digest 5 (MD5) [34], [35], [39].

### 2.3.1 Plaintext Authentication

In plaintext authentication, each participating router must share an authentication key. This key must be specified in each router's configuration. Multiple keys can be specified with some protocols; each key must be identified with a key number. Figure 2.5 illustrates how plaintext authentication is used for routing information.

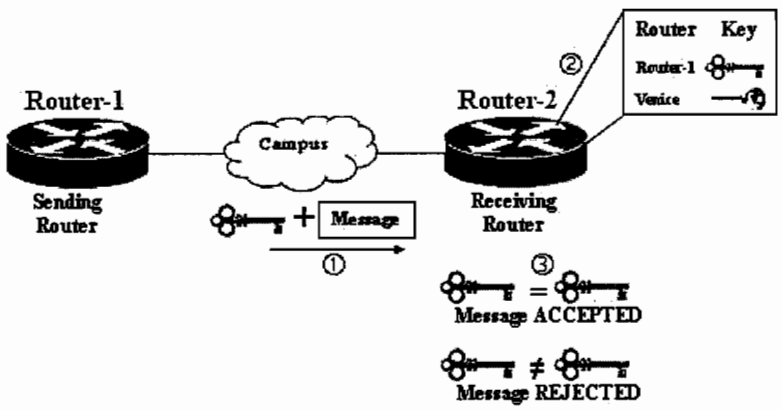


**Figure 2.5: Plaintext Neighbor Authentication.**

In general, from Figure 2.5, when a routing message is sent, the following authentication sequence occurs:

**Step 1** A router sends a routing update with a key and the corresponding key number to the neighbor router. In protocols that can have only one key, the key number is always 0.

**Step 2** The receiving (neighbor) router checks the received key against the same key stored in its own memory.

**Step 3** If the two keys are match, the receiving router accepts the routing message packet. If the two keys do not match, the routing message packet is rejected.

These are Most protocols use plain text authentication: IS-IS, OSPF and RIPv2 [5].

### 2.3.2 MD5 Authentication

MD5 authentication works similarly to plaintext authentication, except that the key is never sent over the wire. Instead, the router uses the combination of a shared secret key, which must be manually preconfigured between the sending and receiving routers, and the routing information as input to the MD5 algorithm to produce a message digest called a hash. The MD5 digest works by creating a 16-byte hash of the routing message combined with a secret key. The 16-byte value is, therefore, message-specific, and modification of the message by an attacker invalidates the 16-byte digest appended to the message. Without the secret key, which is never sent over the wire by the routing protocol, the attacker is unable to reconstruct a valid message. Figure 2.6 illustrates the sequence of events involved for routing protocol authentication for the originating router [14].

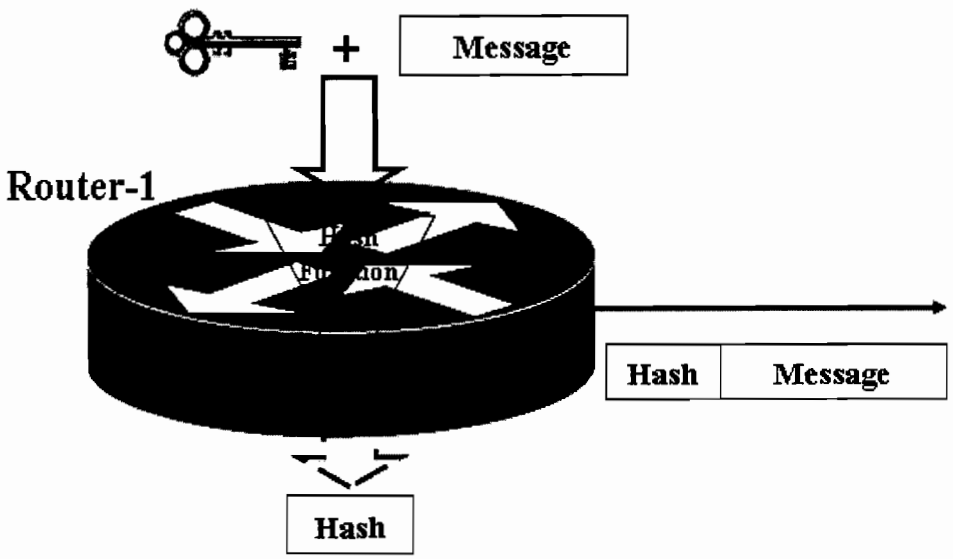


**Figure 2.6: MD5 Neighbor Authentication: Originating Router** [14]

Based on the Figure 2.6, the receiving router takes the routing information, along with its preconfigured shared secret, and uses this as input to the MD5 algorithm to

produce a message digest. If this new digest matches the one that was received, the neighbor is authenticated and the routing information is incorporated into the router's routing information as shown in Figure 2.7.

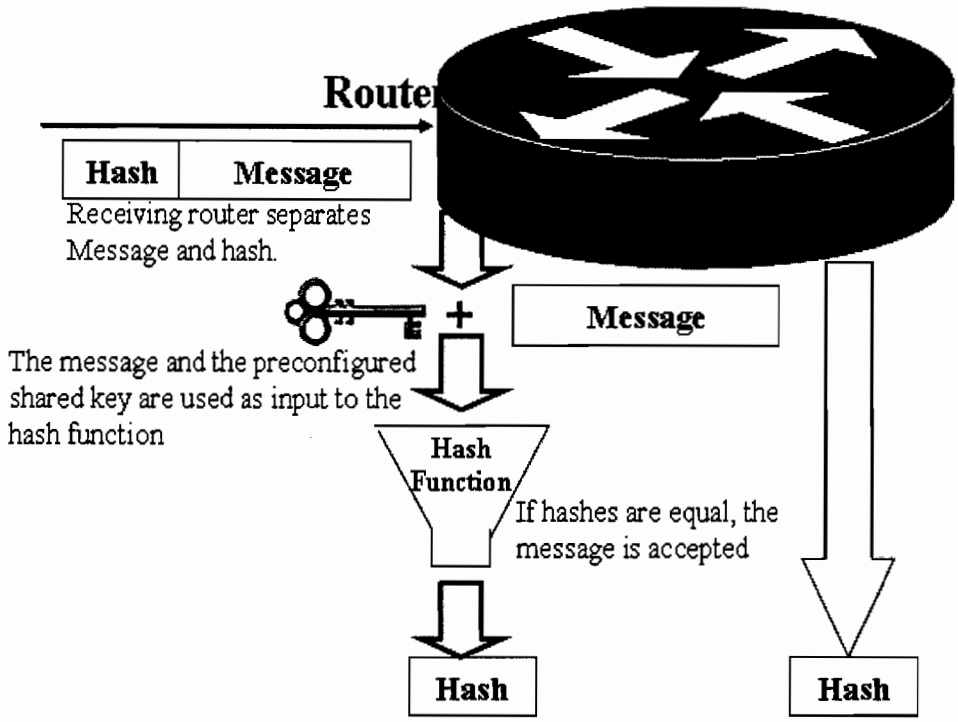


**Figure 2.7: MD5 Neighbor Authentication: Destination Router** [14]

In fact, the MD5 authentication is a more secure place to store the secret key. All calculations can be carried out on the port; which means that the secret key never has to leave the port, avoiding traveling over a secure network. The keys can even be generated on the port itself, which also eliminates any threats to the secret key at initialization. However, the port is vulnerable to loss and theft just like any other hard token. When the secret key is stored in the router port, a password can be used to protect and encrypt it. If the secret key is not password protected, another person could misuse it while the computer is left unattended. However, malicious code entering the computer through the browser could cause damage or theft of the secret key, which is a very serious threat. Various protocols use MD5 authentication are OSPF, RIPv2, BGP, and IP Enhanced IGRP [14], [47], [48].

## 2.4 MD5 Authentication for EIGRP

EIGRP supports only keyed MD5 cryptographic checksums to provide authentication of routing messages. Each key has its own key identifier, which is stored locally. The combination of the key identifier and the interface associated with the message uniquely identifies the authentication algorithm and MD5 authentication key in use. Also, EIGRP MD5 authentication supports multiple keys with lifetimes. Only one authentication packet is sent, regardless of how many valid keys exist. The key numbers are examined in order from lowest to highest, and EIGRP MD5 authentication uses the first valid key it encounters.

EIGRP MD5 authentication ensures that routers accept EIGRP packets only from trusted sources. After the MD5 authentication is configured on an interface, every EIGRP packet sent by a router over that interface is signed with an MD5 fingerprint. Now, every EIGRP packet received over an interface with MD5 authentication configured is checked to verify that the MD5 fingerprint in the packet matches the expected value, making it impossible for the intruder to insert un-trusted routers in the network or send false packets to the routers [40],[41], [48].

MD5 is an algorithm described in RFC 1321 that takes a message, EIGRP packet, and generates 128 bits of hash value, called *message digest* or *fingerprint*, with several properties that make MD5 usable in very secure signature implementations since changing a single bit in the original message changes approximately half of the bits in the MD5 fingerprint. It is almost impossible to generate another message that yields the same MD5 fingerprint; therefore, forging is very hard.

The MD5 value generated from the EIGRP message packet is appended to the EIGRP packet, and the packet is sent to the EIGRP neighbor. The receiving router can verify the integrity of the packet by recalculating the MD5 value and comparing the result with the MD5 fingerprint in the packet. This process does not lead to improved security because an intruder can repeat the steps taken by the originating router and generate forged packets with proper signatures. A secret known only to the sending and receiving router must to be introduced to stop the intruder from generating forged, signed packets. The whole process of secure information

exchange between EIGRP neighbors can be summarized in the following steps graphically presented in Figure 2.8:

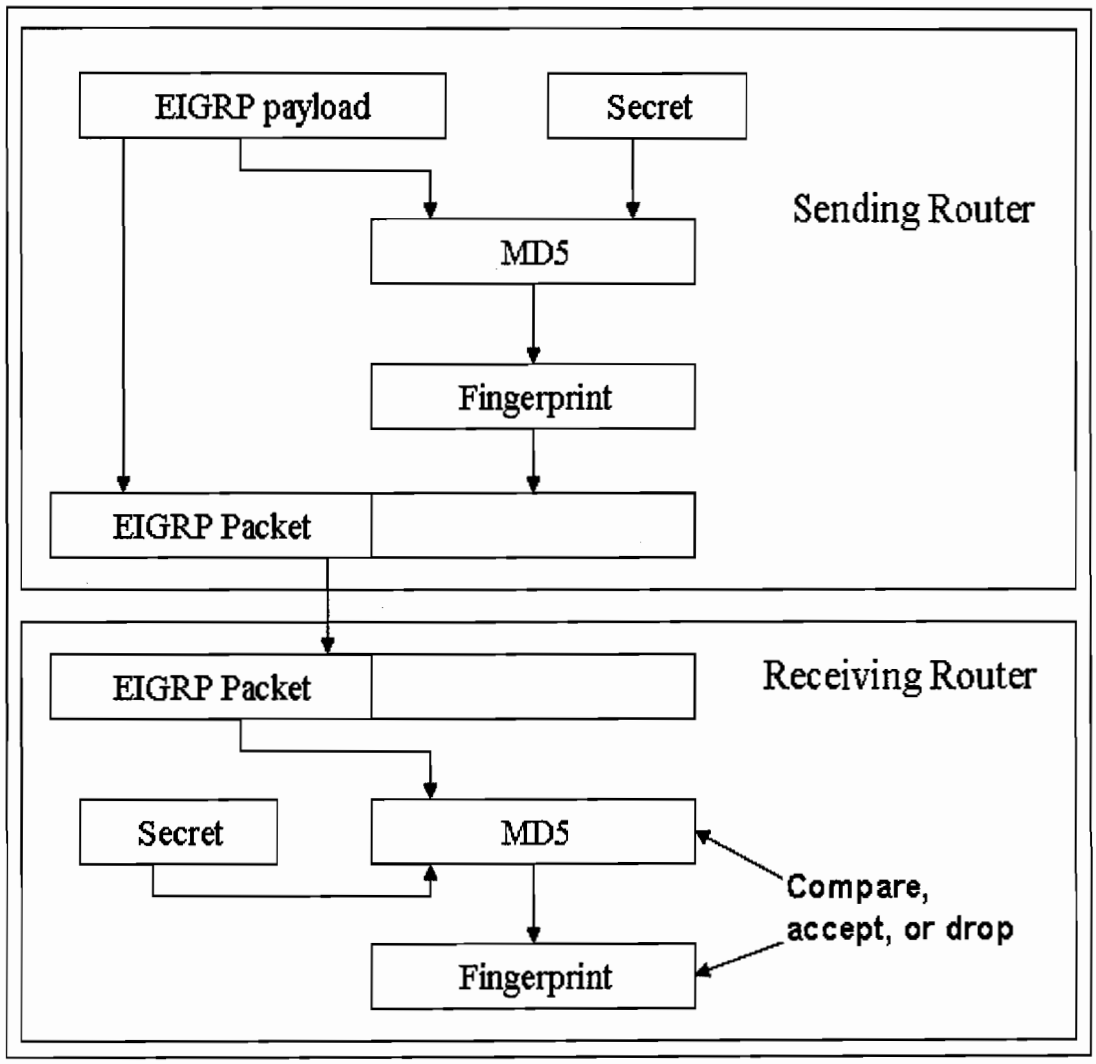


**Figure 2.8: EIGRP MD5 Authentication** [12-p266].

In general, from Figure 2.8, when a routing message is sent, the following authentication sequence occurs:

**Step 1:** The sending router generates EIGRP information to be sent.

**Step 2:** MD5 is computed over EIGRP information and the shared secret.

**Step 3:** The resulting MD5 hash value is appended to the packet and sent to the neighboring router(s). Because the intruder does not know the shared secret, he or she cannot forge the packets.

**Step 4:** The receiving router computes MD5 over received EIGRP information and the shared secret. If the computed MD5 value matches the MD5 fingerprint appended to the packet, the packet is genuine and is accepted for further processing. Packets that do not pass the MD5 fingerprint check are silently dropped [13], [15], [16].

### 2.5 MD5 Authentication for RIPv2

A cryptographic authentication mechanism for RIPv2 is defined in RFC 2082. Keyed MD5 is proposed as the standard authentication algorithm, but the mechanism is intended to be algorithm independent. The basic RIPv2 message format provides for an 8-byte header with an array of 20-byte records as its data content. When keyed MD5 is used, the same header and content are used, except that the 16-byte authentication key field is reused to describe a Keyed Message Digest trailer, as illustrated in Figure 2.9.

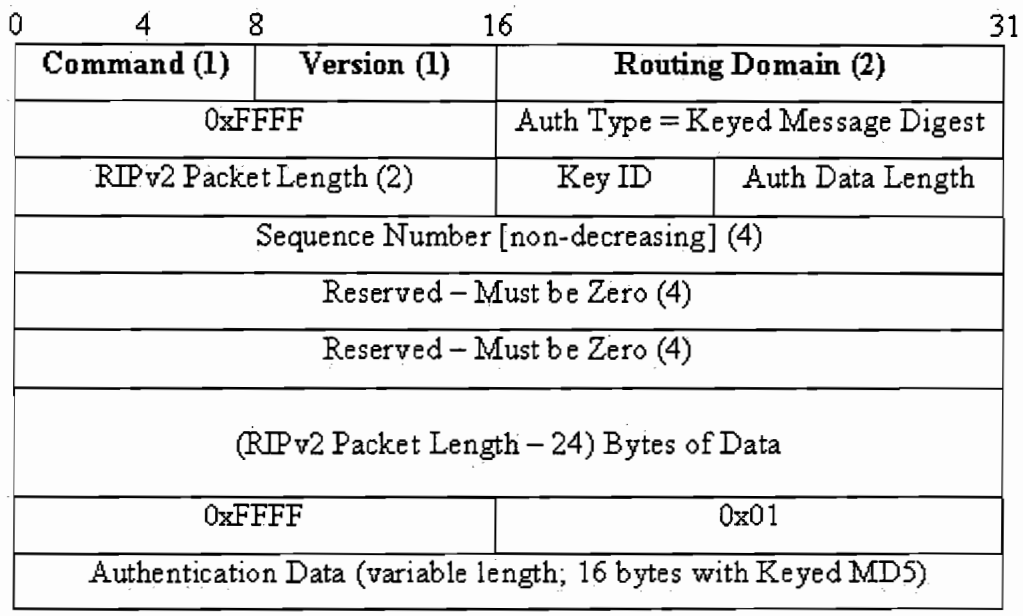


**Figure 2.9: RIPv2 Packet Format Using MD5 Authentication**

RIPv2 MD5 authentication uses the following fields:

- The authentication type is Keyed Message Digest algorithm, indicated by the value 3. 1 and 2 indicate IP route and plaintext password, respectively.

- A 16-bit offset from the RIPv2 header to the MD5 digest. If no other trailer fields are ever defined; this value equals the RIPv2 data length.
- An unsigned 8-bit field that contains the key identifier or key ID. This identifies the key used to create the authentication data for this RIPv2 message. A key is associated with an interface.
- An unsigned, 8-bit field that contains the length in octets of the trailing Authentication Data field. The presence of this field permits other algorithms for instance, Keyed Secure Hash Algorithm (SHA) to be substituted for keyed MD5 if desired.
- An unsigned, 32-bit sequence number. The sequence number is a non-decreasing for all messages sent with the same key ID.
- The authentication data, which is the output of the Keyed Message Digest algorithm. When the authentication algorithm is keyed MD5, the output data is 16 bytes.

During digest calculation, the authentication data is effectively followed by a Pad field and a Length field as defined by RFC 1321. The trailing pad is not actually transmitted, because it is entirely predictable from the message length and algorithm in use. Figure 2.9 illustrates the trailer that is kept in memory and is appended by the MD5 algorithm and treated as though it were part of the message.

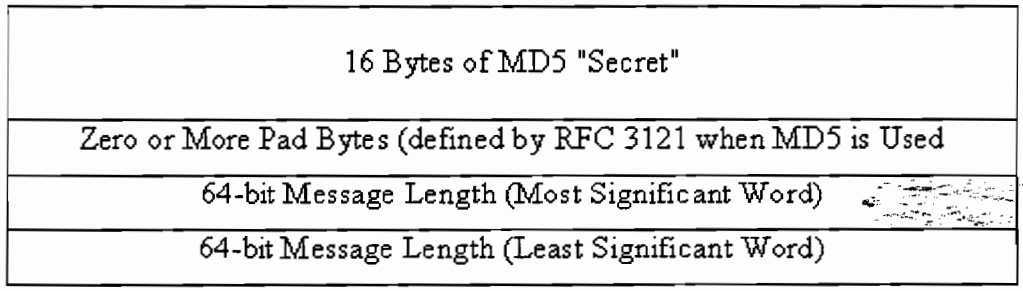


**Figure 2.10: RIPv2 MD5 Trailer**

In addition, the RIPv2 authentication key is selected by the sender based on the outgoing interface. Each key has a lifetime associated with it, and no key is ever used outside its lifetime. Table 2.3 depicts the steps to be carried out at the sending router to generate an authenticated RIP message.

Table 2.3 Steps for Generating an Authenticated RIP Message

| | |
|---|---|
| **Step 1.** | The Authentication Data Offset, Key Identifier, and Authentication Data size fields are appropriately filled in. |
| **Step 2.** | The 16-byte keyed MD5 RIPv2 authentication key is appended to the data. For all algorithms, the RIPv2 authentication key is never longer than the output of the algorithm in use. |
| **Step 3.** | The trailing Pad and Length fields are added and the digest calculated using the indicated algorithm. When keyed MD5 is the algorithm in use, these are calculated per RFC 1321. |
| **Step 4.** | The digest is written over the RIPv2 authentication key. When MD5 is used, this digest is 16 bytes long. |

As we mentioned earlier, there is a trailing pad which is not actually transmitted, because it is entirely predictable from the message length and algorithm in use. When the RIP message is received, however, the following process is reversed as in table 2.4.

Table 2.4 Steps for Retrieving MD5 Digest

| | |
|---|---|
| **Step 1.** | The digest is kept in memory. |
| **Step 2.** | The appropriate algorithm and key are determined from the value of the Key Identifier field. |
| **Step 3.** | The RIPv2 authentication key is written into the appropriate number of bytes starting at the indicated offset. With keyed MD5, 16 bytes are used. |
| **Step 4.** | Appropriate padding is added as needed, and then a new digest is calculated using the indicated algorithm. |

If the calculated digest does not match the received digest, the message is not processed and is discarded. If the neighbor has been heard from recently enough to have feasible routes in the route table and the received sequence number is less than the last one received, the message is also discarded unprocessed. Eventually, the RIPv2 MD5 authentication specification has the following key management requirements:

- Storage of more than one key at the same time, although it is recognized that only one key will normally be active on an interface.
- Associate a specific lifetime and a key identifier with each key.
- Support manual key distribution. For instance, the privileged user manually typing in the key, key lifetime, and key identifier on the router console.
- Keys that are out of date can be automatically deleted by the implementation without requiring human intervention. Manual deletion of active keys can also be supported.

When updating the RIP routers with new keys, a smooth convergence can be ensured if network administrators update all communicating RIPv2 systems with the new key several minutes before the current key expires and several minutes before the new key lifetime begins. The new key should have a lifetime that starts several minutes before the old key expires. This gives time for each system to learn of the new RIPv2 authentication key before that key will be used. It also ensures that the new key will begin being used and the current key will go out of use before the current key's lifetime expires. For the duration of the overlap in key lifetimes, a system may receive messages using either key and authenticate those messages. The key ID in the received message is used to select the appropriate key for authentication.

The specification also recommends that implementations not revert to unauthenticated conditions in the event that the last key associated with an interface expires. It suggests that the router should send a last authentication key expiration notification to the network manager and treat the key as having an infinite lifetime until the lifetime is extended, the key is deleted by network management, or a new key is configured. It is also strongly desirable to use a key management protocol to distribute RIPv2 authentication keys among communicating RIPv2 implementations. However, an integrated key management protocol technique was deliberately omitted from the RIPv2 MD5 specification because at this time of writing specification there does not exist a robust enough key management protocol [15], [34], [47], [48].

## 2.6 MD5 Authentication for OSPF

All OSPF protocol exchanges are authenticated. The OSPF packet header as illustrated in Figure 2.11 includes an Authentication Type field and 64 bits of data for use by the appropriate authentication scheme [3].

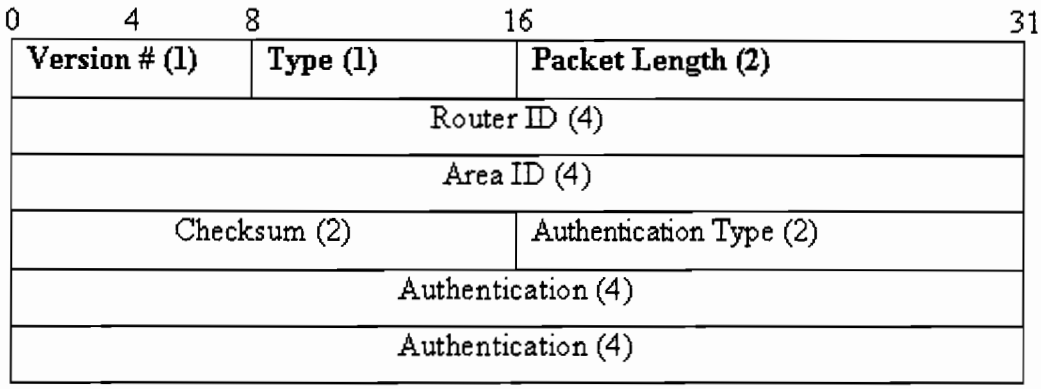


**Figure 2.11: OSPF Packet Header**

Generally, most fields within this common header have obvious meanings. For instance, the version number is set to 2 to indicate OSPFv2 and the type is the OSPF packet type i.e. hello, database description, link-state request, link-state update, and link-state acknowledgment. The packet length is the number of bytes in the packet. The router ID is the IP address selected for identifying the router, and the area ID is the identification of the area where the value zero is reserved for the backbone area. However, it is common practice to choose an IP network number for identifying an area. The checksum is computed over the whole OSPF packet, excluding the 8-byte Authentication field. The Authentication Type field, which is configurable on a router per-interface basis, identifies the authentication algorithm. Three values are defined in the RFC 2328 standard: null authentication, simple password authentication, and plaintext authentication [4], [7], [12].

In null authentication, the routing exchanges over the network or subnet are not authenticated. The 64-bit Authentication field in the OSPF header can contain anything and it is not examined on packet reception. When null authentication is

used, the entire contents of each OSPF packet, other than the 64-bit Authentication field, are check-summed to detect any data corruption and attacks [13], [14], [15].

When using the simple password authentication type, however, a 64-bit field is configured on a per-network basis. All packets sent on a particular network must have this configured value in their OSPF header 64-bit Authentication field. This essentially serves as a clear 64-bit password. In addition, the entire contents of each OSPF packet other than the 64-bit Authentication field are check-summed to detect data corruption. It is worthy to mention here that despite the simple password authentication guards against routers inadvertently joining the routing domain, simple password authentication is vulnerable to passive attacks where anyone with physical access to the network can learn the password and compromise the security of the OSPF routing domain [34], [47], [48].

Eventually, plaintext authentication uses a shared secret key known to all the routers on the network segment. When a sending router builds an OSPF packet, it signs the packet by placing the key as plaintext in the OSPF header. The receiving router then compares the received key against the key in memory. If the keys match, the router accepts the packet. Otherwise, the router rejects the packet [41], [47].

## 2.7 Research Works on Routing Authentication

In a system as large as today's Internet, faults and attacks are inevitable. Given that all Internet based communications rely on a dependable packet delivery service, it is critically important to make network routing protocols highly secured [42]. Consequently, the past decade witnessed a number of research works on this area. For instance, [43] analyzed the security of the Border Gateway Protocol (BGP) routing protocol, and identify a number of vulnerabilities in its design and the corresponding threats. The researcher presented a set of proposed modifications to the protocol which minimize or eliminate the most significant threats. Also, [44] described how to achieve hop integrity in networks that support Internet Protocol (IP). The researcher adopted two famous protocols used in IP networks, namely RIP and OSPF to illustrate how hop integrity can secure the communications between adjacent routers. They argued that every protocol that involves communications

exchanged between adjacent routers can be secured by the deployment of hop integrity in the network.

Traditional routing protocol designs have focused solely on the functionality of the protocols and simplicity assumes that all routing update messages received by a router carry valid information. However, operational experience suggests that hardware faults and operator miss configurations can all lead to invalid routing protocol messages. Thus, the researcher in [26] developed a simple and effective approach to detect invalid routing messages in RIP routing protocol. Their design emphasizes effectiveness, simplicity, low overhead, backward compatibility with the standard RIP protocol, and supports for incremental deployment. Their research work also showed that by carefully exploring the design space of invalid routing messages checking, existing routing protocols can be enhanced with effective fault detection capability as were demonstrated with RIP protocol.

Furthermore, in [27] a survey made on the research efforts over the years aimed at enhancing the dependability of the routing infrastructure. To provide a comprehensive overview of these various efforts, the research work introduced a threat model based on known threats, then sketched out a defense framework, and put each of the existing efforts at appropriate places in the framework based on the faults and attacks against which it can defend. The analysis shows that although individual defense mechanisms may effectively guard against specific faults, no single fence can counter all faults. Thus, a resilient Internet routing infrastructure implies for integrating techniques from cryptographic protection mechanism, statistical anomaly detection, protocol syntax checking, and protocol semantics checking to build a multi-fence defense system. Also, the analysis shows that in order to provide secured neighbor-to-neighbor communication then plaintext passwords and keyed MD5 authentication are needed. Plaintext passwords are vulnerable to eavesdropping, while Keyed MD5 authentication can effectively protect neighbor-to-neighbor protocol exchanges.

In the area of distance vector routing protocols, the research work in [25] proved that such existing protocols are insecure due to the lack of strong authentication and authorization mechanisms and the difficulty, if not impossibility, of validating

routing messages which are aggregated results of other routers. Consequently, the researcher introduced a secure routing protocol, namely Secured-RIP, based on a distance vector approach. In Secured-RIP, a router confirms the consistency of an advertised route with those nodes that have propagated that route. A reputation-based framework is proposed for determining how many nodes should be consulted, flexibly balancing security and efficiency. The threat analysis and simulation results showed that in Secured-RIP, a well-behaved node can uncover inconsistent routing information in a network with many misbehaving nodes assuming no two of them are in collusions, with relatively low extra routing overhead.

## 2.8 Conclusion

In this chapter, we introduced the concept of routing protocol; a protocol that specifies how routers communicate with each other to disseminate information that allows them to select routes between any two nodes on a network. We also differentiate between two types of routing protocols: Distance Vector Routing and Link State Routing. Basically, Distance Vector protocols, such as RIP and EIGRP, determine best path on how far the destination is, while Link State protocols, such as OSPF, are capable of using more sophisticated methods taking into consideration link variables, such as bandwidth, delay, reliability and load.

Most routing protocols incorporate neighbor authentication to protect the integrity of the routing domain and prevent a legitimate router from accepting and then using unauthorized, malicious, or corrupted routing messages that may compromise the security or availability of the network. Typically, there are two types of neighbor authentication: plaintext authentication and cryptographic authentication mainly using the Message Digest 5 (MD5).

In plaintext authentication, each participating router must share an authentication key. This key must be specified in each router's configuration. On the other hand, MD5 authentication works similarly to plaintext authentication, except that the key is never sent over the wire. Instead, the router uses the combination of a shared secret key, which must be manually preconfigured between the sending and receiving

routers, and the routing information as input to the MD5 algorithm to produce a message digest called a hash.

For EIGRP authentication, only keyed MD5 cryptographic checksums is supported to provide secured EIGRP routing messages. With this scheme, the key numbers are examined in order from lowest to highest, and EIGRP MD5 authentication uses the first valid key it encounters. EIGRP MD5 authentication ensures that routers accept EIGRP packets only from trusted sources through verifying the integrity of the EIGRP packet by recalculating the MD5 value and comparing the result with the MD5 fingerprint in that packet.

The basic RIPv2 message format provides an 8-byte header with an array of 20-byte records as its data content. When keyed MD5 is used, the same header and content are used, except that the 16-byte authentication key field is reused to describe a Keyed Message Digest. In addition, the RIPv2 authentication key is selected by the sender based on the outgoing interface. Each key has a lifetime associated with it, and no key is ever used outside its lifetime. Eventually, if the calculated digest does not match the received digest, the message is not processed and is discarded.

All OSPF protocol exchanges are authenticated using three main types: null authentication, simple password authentication, and plaintext authentication. When null authentication is used, the entire contents of each OSPF packet, other than the Authentication field, are check-summed to detect any data corruption and attacks. The simple password authentication guards against routers inadvertently joining the routing domain. However, simple password authentication is vulnerable to passive attacks where anyone with physical access to the network can learn the password and compromise the security of the OSPF routing domain. Finally, plaintext authentication uses a shared secret key known to all the routers on the network segment to provide a secured OSPF messages.

# Chapter 3

# Research Methodology

## 3.1 Introduction

In this chapter, we present in details the research methodology we adopt for evaluating different routing protocols performance in the context of: secured Md5 Authentication and non-secured situations. In general, our research method can be characterized by the sequence diagram shown in Figure 3.1.

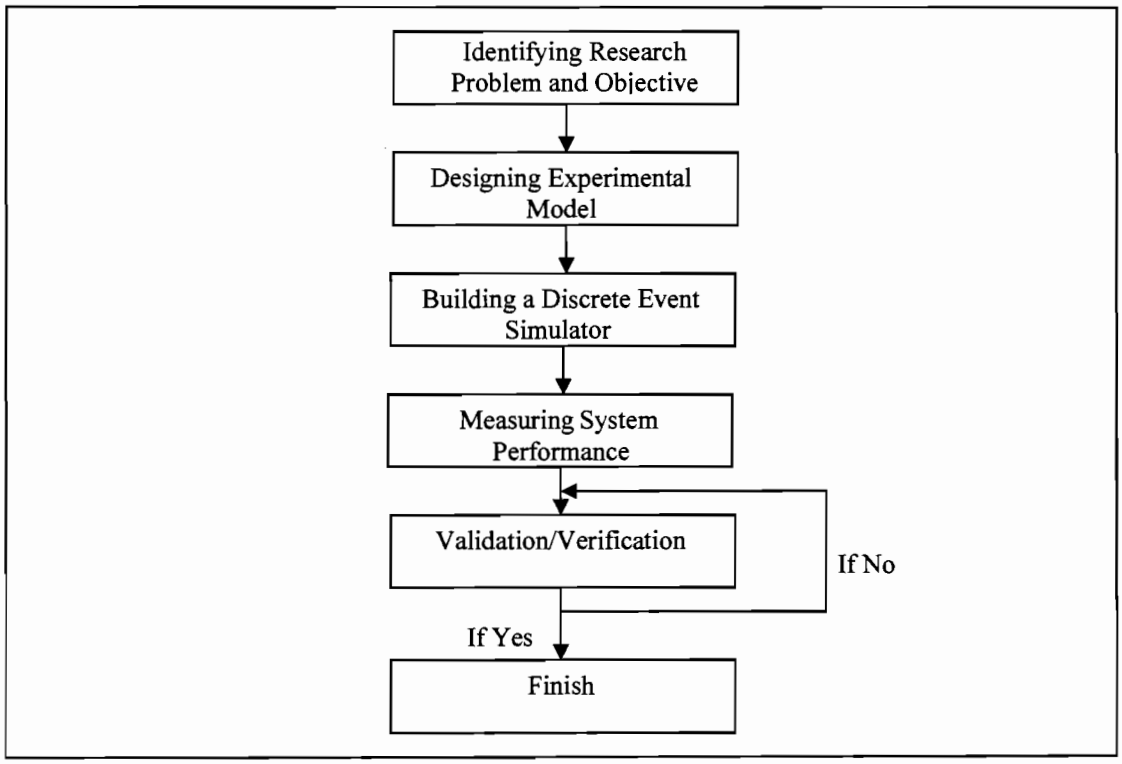


**Figure 3.1: Research Method** [51]

As the figure shows, our methodology starts by identifying the major research problem and objective. To do so, an extensive literature review on authenticated routing messages was carried out in Chapter 2. The literature revealed that two types of routing authentication are commonly used: plaintext authentication and cryptographic authentication typically using the Message Digest 5 (MD5). With the desirable property for the MD5 authentication to eliminate most threats and data corruption, MD5 was widely adopted. Generally speaking, to secure a routing

protocol it is required that important routing information be authenticated between neighboring routers. In a routing protocol context, when a router sends out a routing message, the identity of the originator of the information should be validated in terms of integrity and confidentiality. Integrity services ensure that the data being transmitted is consistent with the data being received. Confidentiality service provides privacy of routing message, which uses encryption to prevent others from knowing what the routing message is. Consequently, our research's fundamental objective, to ensure such authentication, will be achieved in the next step of our methodology by designing an experimental model which will enable us to evaluate the performance of different routing protocols. The design will shows the needed networking devices to be used in such experimental model [48].

Once the experimental model is built then it must be evaluated with the proper performance measure of interest. However, since we have two alternative techniques, mathematical and physical, to evaluate such model then we will adopt the second performance evaluation technique. However, since we need an input data for this physical model then our next research methodology step will start by building a discrete event simulator to generate such input data. The discrete-event simulator with client and server sides is constructed in such a way that the traffic generated by client will be passed to the server via the experimental physical model.

Eventually, to measure and evaluate the performance of this system, we set three performance measures of interest. First, we will estimate the expected packet average delay time, second we will measure the variation of such average delay known as jitter, and third we will measure the variation of overhead for all routing protocols [49].

### 3.2 Identifying Research Problem

Generally speaking, to secure a routing protocol it is required that important routing information be authenticated between neighboring routers. Those various kinds of attacks actually take advantage of the lack of authenticity, integrity or confidentiality. Authentication services are primarily concerned with the providing assurances about the identity of an entity. In a routing protocol context, when a router sends out a

routing message, the identity of the originator of the information should be able to be validated. Integrity services ensure that the data being transmitted is consistent with the data being received. Confidentiality service provides privacy of routing message, which uses encryption to prevent others from knowing what the routing message is.

Moreover, the current state of the ability in protecting the routing infrastructures relies on so-called best practices, which include various simplistic techniques such as passwords, TCP, authentication, route filters, and private addressing to ease the most basic vulnerabilities and threats [5], [6]. Authentication occurs when two neighboring routers exchange routing information and ensures that the receiving router incorporates into its tables only the route information that the trusted sending neighbor really intends to send. It prevents a genuine router from accepting and then using unauthorized, malicious, or corrupted routing updates that may compromise the security or availability of the network. Such a compromise would lead to rerouting of traffic, or a denial of service.

In this research work, we will evaluate the performance of EIGRP, RIPv2 and OSPF routing protocols in the context of: secured MD5 Authentication and non-secured situations. To meet this goal, a test-bed network model of four Cisco routers will be employed. A traffic generation and analysis tools will be developed to generate traffic data and to measure the average delay time, jitter and overhead as our performance measures of interests. Specifically, in the experiment model, a Java-based Object-oriented discrete-event program with both client and server will be implemented at the end nodes of the test-bed network model. The network traffic, as TCP packets, will be directed from the client side to the server side which eventually calculates the major performance measures. The generation of these packets follows the Markov Poisson Process (MPP), which is a doubly stochastic Poisson Process whose rate varies according to a Markov Process. The MPP can be viewed as a superposition of latent Poisson processes, which can be expressed as a non-homogeneous discretely indexed Hidden Markov Model (HMM) by partitioning time into intervals between observed events. The resultant traffic model is an exponentially distributed ON/OFF traffic where the client sends bulk traffic only during the ON periods [40] [41].

## 3.3 Designing the Experimental Model

In this research, we intend to design a physical model for carrying out our research experiment. Specifically, the designed model will be used to evaluate different routing protocols for both secured MD5 Authentication and non-secured situations. Accordingly, to achieve such objective, our proposed model should satisfy the following research questions:

1. Can we demonstrate that our proposed design model is enough robust to handle the input data generated by the client?

2. Can we guarantee that our proposed design model is capable of achieving our main research objective?

3. Can we ensure that the proposed design model is of minimum overhead?

4. Can we ensure that the proposed design model is highly synchronized within its neighboring routers and client/server?

## 3.4 Selecting Evaluation Technique

A system is defined to be a collection of entities, e.g. people or machines, which act and interact together toward the accomplishment of some logical end [50]. In practice, what is meant by the system depends on the objectives of a particular study. The collection of entities that comprise a system for one study might be only a subset of the overall system for another. However, the state of a system can be defined as the collection of variables necessary to describe that system at a particular time, relative to the objectives of a study.

In general, systems can be categorized into two types, discrete and continuous. A discrete system is one for which the state variables change instantaneously at separated points in time. A continuous system is one for which the state variables change continuously with respect to time. Few systems in practice are wholly discrete or wholly continuous; but since one type of change predominates for most

systems, it will usually be possible to classify a system as being either discrete or continuous. However, at some point in the lives of most systems, there is a need to study them to try to gain some insight into the relationships among various components, or to predict performance under some new conditions being considered. Figure 3.2, maps out different ways in which a system might be studied. Hence, the ways for studying a particular system can be categorized by the following [51]:

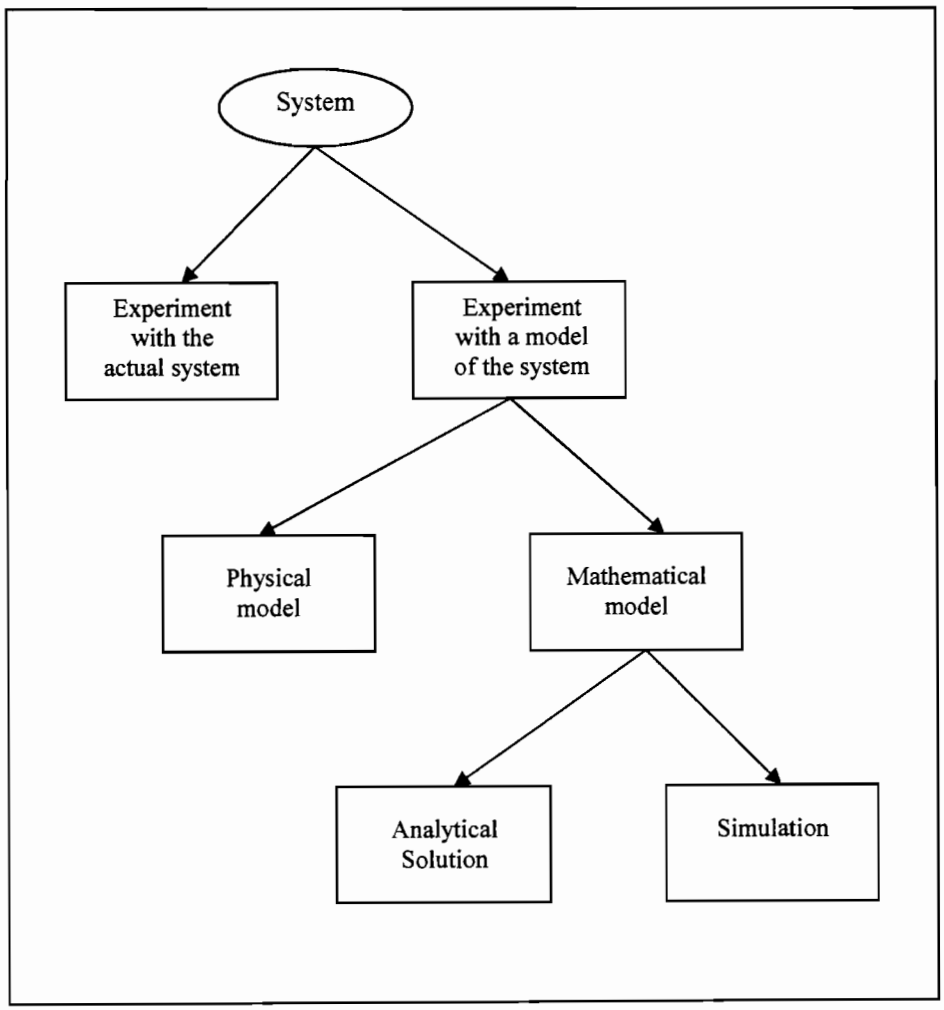


**Figure 3.2: Ways of Studying Systems**

- *Experiment with the Actual System vs. Experiment with a Model of the System.* If it is possible (and cost-effective) to alter the system physically and then let in operate under the new conditions, it is probably desirable to do so, for in this case there is no question about whether what we study is valid. However, it is rarely feasible to do this, because an experiment would often be too costly or too disruptive to the system. More graphically, the system might not even exist,

but we nevertheless want to study it in its various proposed alternative configurations to see how it should be built in the first place; such as a proposed communication network, or a strategic unclear weapons system. For these reasons, it is usually necessary to build a model as a representation of the system and study it as a surrogate of the actual system.

- *Physical Model vs. Mathematical Model.* It has been found useful to build physical models (also called iconic models) to study engineering or management system. But the vast majority of models built for such purposes are mathematical, representing a system in terms of logical and quantitative relationships that are then manipulated and changed to see how the model reacts, and thus how the system would react-if the mathematical model is a valid one.

- *Analytical Solution vs. Simulation.* Once the mathematical model has been built, it must be then examined to see how it can be used to answer the questions of interest about the system it is supposed to represent. If the model is simple enough, it may be possible to work with its relationships and quantities to get an exact analytical solution. This is very simple, closed-form solution obtainable with just paper and pencil, but some analytical solutions can become extraordinary complex, requiring vast computing resources. If an analytical solution to a mathematical model is available and is computationally efficient, it is usually desirable to study the model in this way rather than via a simulation. However, many systems are highly complex, so that valid mathematical models of them are themselves complex. In this case, the model must be studied by means of simulation, i.e. numerically exercising the model for the inputs in question to see how they affect the output measures of performance. Simulation is one of the most widely used operations-research and management science techniques, if not the most widely used.

For our research work, we choose to experiment with a physical model of the system rather than with its mathematical one; since experiment with the physical model is more useful and almost less costly. Once the proposed physical model is built then it must be evaluated with the proper performance measures of interest.

However, since we need an input data for this physical model then we will need to build a simulator tool to generate such input data. In fact, simulation is one of the most widely used operation-research and management science techniques. It is defined as the imitation of the operation of a real-world process or system over time. Thus, simulation modeling can be used both as an analysis tool for predicting the effect of changes to existing systems, and as a design tool to predict the performance of new systems under varying sets of circumstances. Consequently, Simulation models can be classified based on seven different dimensions [52]:

- *Static vs. Dynamic Models.* A static simulation model is a representation of a system at a particular time, or one that may be used to represent a system in which time simply plays no role. On the other hand, a dynamic simulation model represents a system as it evolves over time, i.e. the system state changes with time.

- *Deterministic vs. Stochastic (Probabilistic) Models.* If a simulation model does not contain any probabilistic (i.e. random) components, it is called deterministic; a complicated system of differential equations describing such a model. In deterministic models, the output is determined once the set of input quantities and relationships in the model have been specified; even though it might take a lot of computer time to evaluate what is it. Many systems, however, must be modeled as having at least some random input components, and this give rise to stochastic simulation models. For example, most queuing systems are modeled stochastically. Stochastic simulation models produce output that is itself random and must be therefore treated as only an estimate of the true characteristics of the model.

- *Continuous vs. Discrete Models.* A model in which the system state variables are defined at all times is called continuous-time model. On the other hand, if the system state variables are defined only at particular instants in time, the model is called discrete-time model.

In our research, the specific nature of the developed simulator is considered as a Dynamic Stochastic Discrete event simulation. A dynamic simulation model

represent a system as it evolves over time, stochastic is defined as the simulation model contains some probabilistic such as random input components. Finally, discrete event simulation concerns the modeling of a system as it evolves over time by a representation in which the state variables change instantaneously at separate points in time [50]. These points in time are the ones at which an event occurs, where an event is defined as an instantaneous occurrence that may change the state of the system.

Because of the dynamic nature of discrete-event simulation models, we must keep track of the current value of simulated time as the simulation proceeds, and also need a mechanism to advance simulated time from one value to another. For that reason, a variable, called simulation clock must be defined in the simulation model to give the current value of simulated time. The unit of time for that simulation clock is never stated explicitly when a model is written in a general-purpose language, and it is assumed to be in the same units as the input parameters.

Historically, two principal approaches have been suggested for advancing the simulation clock: *next-event time advanced and fixed-increment time advance.* However, the first approach is used by all major simulation software and by most people coding their model in a general-purpose language, while the second is a special case of the first. With Fixed-Increment Time Advance approach, the simulation clock is advanced in increments of exactly same time units. After each update of the clock, a check is made to determine if any events should have occurred during the previous interval of length. If one or more events were scheduled to have occurred during this interval, these events are considered to occur at the end of the interval and the system state are updated accordingly. The primary use of this approach appears to be for systems where it can reasonably be assumed that all events actually occur at fixed intervals. On the other hand, with Next-Event Time Advanced approach, the simulation clock is initialized to zero and the times of occurrence of future events are determined. The simulation clock is then advanced to the time of occurrence of the first of these future events, at which point the state of the system is updated to account for the fact that an event has occurred.

## 3.5 Modeling Arrival Process

In many simulations it is needed to generate a sequence of random points in time $0 = t_0 \leq t_1 \leq t_2 \leq \ldots \leq t_n$, such that the *i*th event of some kind occurs at time $t_i$ ($i = 1, 2 \ldots n$) and the distribution of the event times $\{t_i\}$ follows some specified form.

In this research, two models of arrival processes are considered: first, the Poisson process, which is an arrival process for which its values are IID (Identically and Independently Distributed) exponential random variables. The Poisson process is probably the most commonly used model for the arrival process of customers to a queuing system. Second, the Batch arrival, which is an approach for modeling arrival processes where each event is actually an arrival of a group of packets.

*Poisson Processes*: The stationary Poisson process with rate $\lambda > 0$, has the property that the inter-arrival times $A_i = t_i - t_{i-1}$ (where $i = 1, 2 \ldots$) are IID exponential random variables with common mean $1/\lambda$. Thus, we can generate the $t_i$'s recursively as follow:

1. Generate $U \sim U(0, 1)$ independent of nay previous variants.
2. Return $t_i = t_{i-1} - (1/\lambda) \ln U$.

This algorithm can be easily modified to generate any arrival process where the inter-arrival times are IID random variables, whether or not they are exponential.

*Batch Arrivals*: In this approach we consider an arrival process *i*th batch of customers arrives at time $t_i$ and the number of customers in this batch is a discrete random variable $B_i$. If we assume that the $B_i$'s are IID and, in addition, are independent of the $t_i$'s. Then a general recursive algorithm for generating this arrival process is as follow:

1. Generate the next arrival time $t_i$.
2. Generate the discrete random variants $\boldsymbol{B_i}$ independently of any previous $\boldsymbol{B_j}$'s and also independently of $t_1, t_2 \ldots t_i$.
3. Return the information that $B_i$ customers are arriving at time $t_i$.

## 3.6 Measuring System Performance

To measure and evaluate the performance of this system, we will look at estimates of two quantities. First, we will estimate the expected packet average delay of the $n$ packets completing their delays during the simulation; we denote this quantity by $d(n)$. Since the actual average delay observed of the $n$ packets, on a given run of the simulation, depends on the inter-arrival and service-time random variable observations, then the average delay on a given run of the simulation is properly regarded as a random variable itself. As a result, the estimation of this measure is expressed by the expected value of this random variable.

From a single run of the simulation resulting in packet delays $D_1, D_2, \ldots D_n$, an obvious estimator of $d(n)$ is

$$\hat{d}(n) = \frac{\sum_{i=1}^{n} D_i}{n} \tag{3.1}$$

The second and final output measure and evaluate of performance for this system is a measure of jitter which is defined as a variation in the delay of received packets. At the sending side, packets are sent in a stream with the packets spaced evenly apart. Due to network congestion, improper queuing, or configuration errors, this steady stream can become lumpy, or the delay between each packet can vary instead of remaining constant. Hence, from multiple run of the simulation resulting in packet's average delays $D_1\hat{}, D_2\hat{}, \ldots D_n\hat{}$, we can estimate the variation of such delays.

## 3.7 Conclusion

In this chapter, we presented in details the research methodology we adopt for evaluating different routing protocols performance in the context of: secured and non-secured situations. We started our methodology by identifying the main research problems related to authenticate routing protocol messages. And consequently, we defined our proposed experimental model, which will be used in studying the performance of secured versus non-secured routing protocol messages.

On determining the performance evaluation techniques to be adopted for carrying out this research, we decided to experiment with a physical model of the system rather than with its mathematical one. However, since we need an input data for this physical model then we outline that our next step of research approach is to build a simulator tool to generate such input data.

The specific nature of the developed simulator is considered as Dynamic Stochastic Discrete event simulation. A dynamic simulation model represent a system as it evolves over time, stochastic is defined as the simulation model contains some probabilistic such as random input components, and discrete event concerns the modeling of a system as it evolves over time by a representation in which the state variables change instantaneously at separate points in time.

Finally, to measure and evaluate the performance of this system, we looked at estimates of three quantities. First, we will estimate the expected packet average delay time of the system packets completing their delays during the simulation. The second output measure of performance for this system is a measure of jitter which is defined as a variation in the delay of received packets. The third output measure of performance for this system is a measure of the variation of overhead for all routing protocols.

# Chapter 4

## The Test-bed Network Model

### 4. 1 Introduction

In this chapter, we are presenting in the details of the real experiment network model we adopt for evaluating the performance of different routing protocols in the context of MD5 Authentication secured and non-secured situations. Accordingly, we will start our chapter by explaining our test-bed network model and its features before we strictly describe the MD5 authentication technique and our Java-based Object-Oriented discrete-event simulator.

### 4. 2 Test-Bed Network Model (Cisco Routers)

The real proposed test-bed network model consists of four Cisco Routers 1721 Modular Access Router, Client/Server, Transmission media connections and a Java client-server program for generating, monitoring traffic and reporting results is presented as part of this work. The Java-based Object-oriented discrete-event simulator with both client and server is implemented at each end node of the network model. The network traffic, namely TCP packets, is directed from the client to the server, which calculates the major performance measures, especially the average delay time, jitter and overhead of the TCP packets. Figure 1 illustrates the real proposed test-bed network model [53], [54].

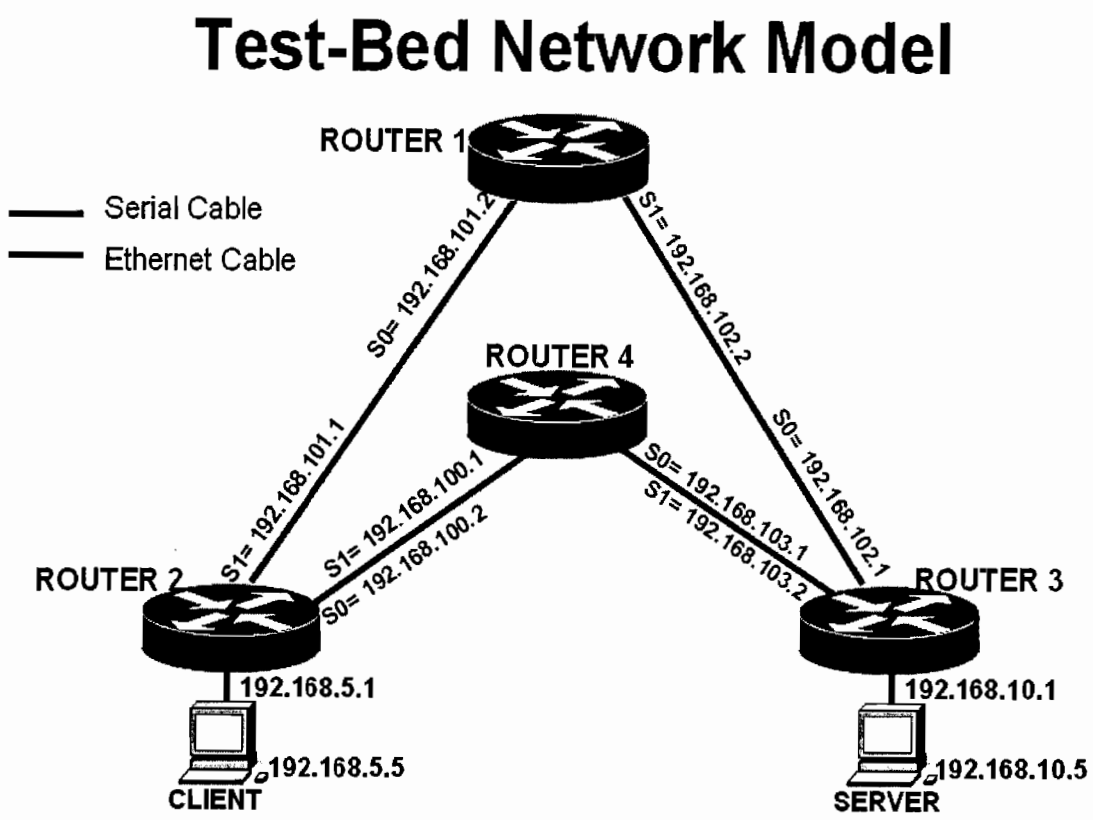


**Figure 4.1: Test-bed Network Model**

### 4. 2. 1 Cisco Routers 1721 Modular Access Router

The Cisco router 1721 Modular Access Router has good characteristics to use as it is reliable, dependable, and secured in both internet and network access through various high-speed WAN access technologies. The Cisco 1700 Series offers a comprehensive suite of integrated security capabilities with wire-speed IP Security Virtual Private Networks (VPN), stateful firewall protection, and intrusion detection. It also offers a migration path to voice-over-IP and IP telephony services through a converged data and voice network that offers call processing and Quality of Service (QoS) guarantees. Figure 4.2 shows front and back sides for the Cisco 1721 Router [54].

**Figure 4.2: Cisco Routers 1721 Modular Access Router**

Indeed, Cisco router 1721 is an ideal for enterprise branch offices small and medium-sized businesses, the Cisco 1700 Series modular design provides the flexibility to meet demanding and evolving business requirements by offering high-speed broadband and leased-line access, comprehensive security, and multi-service data and voice integration [53].

### 4. 2. 2 Cisco Router 1721 Key Feature

The Cisco 1721 Modular Access Router is designed to help organizations embrace the productivity benefits of secured applications. The Cisco 1721 router enables such benefits by delivering secure Internet, intranet, and extranet access with (VPN) and firewall technology. Table 4.1 shows the key features of Cisco Routers 1721 Modular Access Router.

Table 4.1: Key Features of Cisco Routers 1721

| Feature | Description |
|---|---|
| One Fast Ethernet (10/100BASE-TX) port | Operates in full- or half-duplex mode. Supports auto-sensing for 10- or 100-Mbps operation. Supports IEEE 802.1Q VLAN encapsulation. |
| Two Cisco WAN interface card, (WIC) slots | Supports a combination of any two of the following WIC's: ISDN BRI, 56-kbps DSU/CSU, FT1/T1 DSU/CSU, high-speed serial, dual-serial, ADSL, and Ethernet. |
| One Console port | Supports router configuration and management with a directly-connected terminal or PC. Supports up to 115.2 kbps. |

| | |
|---|---|
| One Auxiliary port | Supports modem connection to the router, which can be configured and managed from a remote location. Supports up to 115.2 kbps. |
| VPN hardware-assisted 3DES, encryption module | Provides IPSEC DES and 3DES hardware encryption. |
| SNMP support | Router can be managed over a network using Simple Network Management Protocol (SNMP). |
| Support for Cisco IOS software version 12.4 features. | Supports IP, IPX, AppleTalk, IBM, Open Shortest Path First (OSPF), NetWare Link Services Protocol (NLSP), Resource Reservation Protocol (RSVP), encryption, network address translation, and the Cisco IOS Firewall Feature Set |

[54], [55]

### 4. 2. 3 Front Panel LED's

This subsection looks for the router front panel LED's, which are shown in Figure 4.3 and its detailed description, is illustrated in Table 4.2.

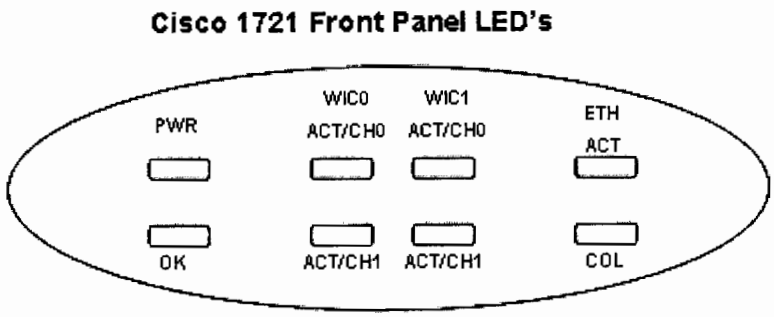


**Figure 4.3: Front panel of Router 1721**

Table 4.2: Cisco Routers 1721 Front Panel LEDs Description

| LED | Indication | Description |
|---|---|---|
| PWR | Green /Off | Power is supplied to the router / The router is not powered on. |
| OK | Green / Off | The router has successfully booted up and the software is functional. This LED blinks during the |

| | | |
|---|---|---|
| | | power-on self-test (POST) / The router has not successfully booted up. |
| WIC 0 ACT/CH0 | Green | Serial and DSU/CSU cards—Blinks when data is being sent to or received from the port on the card in the WIC0 slot |
| WIC 0 ACT/CH1 | Green | Serial and CSU/DSU cards—Remains off |
| WIC 1 ACT/CH0 | Green | Serial and DSU/CSU cards—Blinks when data is being sent to or received from the port on the card in the WIC1 slot |
| WIC 1 ACT/CH1 | Green | Serial and CSU/DSU cards—Remains off |
| ETH ACT | Green | Blinks when there is network activity on the Ethernet port |
| ETH COL | Yellow | Blinks when there are packet collisions on the local Ethernet network |

[54], [55]

### 4. 2. 4 Back Panel Ports and LED's

In this subsection, we describe the router back panel ports and LED's, which are shown in Figure 4.4 and its detailed description is illustrated in Tables 4.3 and 4.4.

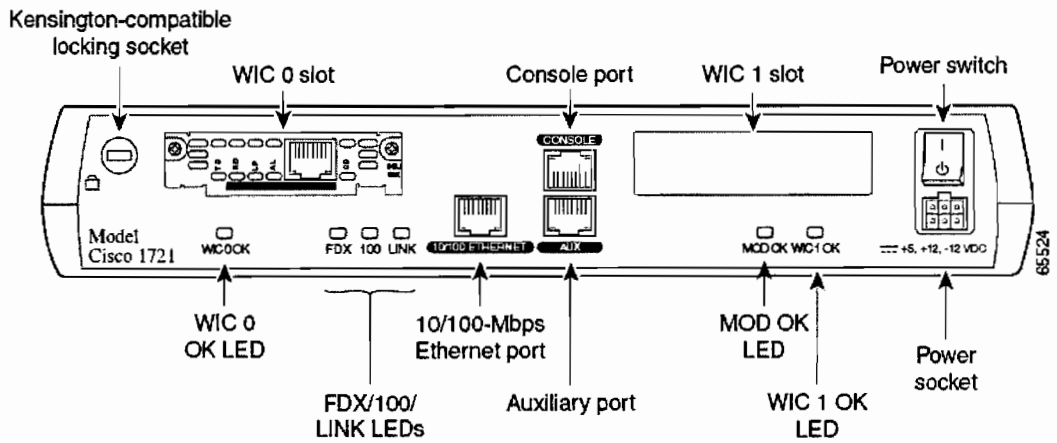


**Figure 4.4: Back panel of Router 1721**

Table 4.3: Cisco Routers 1721 Back Panel Ports Description

| Connector/Slot | Label/Color | Description |
|---|---|---|
| Ethernet port | 10/100 ETHERNET (yellow) | Connects the router to the local Ethernet network through this port. This port autosenses the speed (10 Mbps or 100 Mbps) and duplex mode (full- or half-) of the device to which it is connected and then operates at the same speed and in the same duplex mode. |
| Auxiliary port | AUX (black) | Connects to a modem for remote configuration with Cisco IOS software. Supports up to 115.2 kbps. |
| Console port | CONSOLE (blue) | Connects to a terminal or PC for local configuration using Cisco IOS software. Supports up to 115.2 kbps. |
| WIC slot0 (WIC0) | No label | Supports one Cisco *WAN Interface Cards* (WIC). |
| WIC slot1 (WIC1) | No label | Supports one Cisco *WAN Interface Cards* (WIC) |

Table 4.4: Cisco Routers 1721 Back Panel LEDs Description

| LED Label | Color | Description |
|---|---|---|
| WIC0 OK | Green | On when a WIC is correctly inserted in the card slot. |
| FDX | Green | On solid—Ethernet port is operating in full-duplex mode. Off—Ethernet port is operating in half-duplex mode |
| 100 | Green | On solid—Ethernet port is operating at 100 Mbps.<br>Off—Ethernet port is operating at 10 Mbps |
| LINK | Green | On when the Ethernet link is up. |
| MOD OK | Green | On when the VPN hardware encryption module is installed and recognized by the IOS. |
| WIC1 OK | Green | On when a WIC is correctly inserted in the card slot. |

### 4. 2. 5 Physical Interfaces

The interfaces on a router provide network connectivity to the router. The console and auxiliary ports are used for managing the router. Routers also have ports for LAN and WAN connectivity. The LAN interfaces usually include Ethernet, Fast Ethernet, while the serial interfaces are used for WAN connectivity. Specifically, for Cisco Routers 1721, the following ports are identified:

1. One 10/100BASE-TX Fast Ethernet port (RJ-45)
   - Automatic speed detection and duplex negotiation
   - IEEE 802.1Q VLAN routing

2. Two WAN interface card slots
   - Supports any combination of two WAN interface cards.

3. One auxiliary (AUX) port
   - RJ-45 jack with EIA/TIA-232 interface
- Asynchronous serial data terminal equipment (DTE) with full modem controls Carrier Detect, data set ready (DSR), Request To Send (RTS), Clear To Send (CTS)
   - Asynchronous serial data rates up to 115.2 kbps
4. One console port
   - RJ-45 jack with EIA/TIA-232 interface
   - Asynchronous serial DTE
   - Transmit/receive rates up to 115.2 kbps (default 9600 bps, not a network data port)
   - No hardware handshaking such as RTS/CTS

5. One internal expansion slot for support of hardware-assisted services such as VPN encryption (up to T1/E1 performance). [53], [54], [55]

### 4. 2. 6. LAN Adapter

A LAN adapter is a device used to allow a router to interface with a network. Many routers may have some of LAN adapter already installed. Most networks that are used in an office or home environment are known as Local Area Networks (LANs). This type of network is one used over a limited geographic area. A LAN adapter is simply one that is able to access this type of network. A LAN adapter can be used with a wireless or wired network, it is important to understand what type of network and connection is needed. In most cases, a wireless LAN adapter cannot be used for a wired network and vice versa. In many cases, a wired LAN adapter is used for Ethernet connections, one of the fastest and most reliable forms of wired networks. Because of their performance and security, they are often used in office or business environments. The following port is identified:

- One 10/100BASE-TX Fast Ethernet port (RJ-45)
  - Automatic speed detection
  - Automatic duplex negotiation
  - IEEE 802.1Q VLAN routing (Cisco 1721 only) [55]

### 4. 2. 7 WAN Adapters

The dual serial port WAN Interface Card (WIC−2A/S) provides higher levels of serial port density for a single WIC and is supported on the Cisco 1700, 2600, and 3600 series. The low serial speed WIC−2A/S supports up to 128 Kbps synchronous or 115.2 Kbps asynchronous serial links. Each port on a WIC is a different physical interface and can support different protocols such as Point−to−Point Protocol (PPP) or Frame Relay and Data Terminal Equipment/Data Communications Equipment (DTE/DCE). This WIC supports mixed asynchronous and synchronous operation on a single card as well. The dual-serial port WAN interface cards (WICs) for the Cisco 1700 series feature Cisco`s new, compact, high-density Smart Serial connector to support a wide variety of electrical interfaces when used with the appropriate transition cable. This includes V.35, RS−232, RS−449, RS−530, RS−530A in male and female versions for both DTE and DCE devices. This feature provides easy configuration and reconfiguration as network requirements change, without the need

of purchasing a different serial interface card. Here are the specifications of WAN interface port:

- Synchronous serial interfaces on serial WAN interface cards
  - Interface speed: up to 2.0 Mbps (T1/E1).
  - Synchronous serial protocols: Point-to-Point Protocol (PPP), High-Level Data Link Control (HDLC), Link Access Procedure, Balanced (LAPB), IBM Systems Network Architecture (SNA).
  - Synchronous serial WAN services: Frame Relay, X.25, SMDS.
  - Synchronous serial interfaces supported on the WIC-1T, WIC-2T, and WIC-2A/S cards: V.35, EIA/TIA-232, EIA/TIA-449, X.21, EIA-530

- Asynchronous serial interfaces on serial WAN interface cards
  - Interface speed: up to 115.2 kbps
  - Asynchronous serial protocols: PPP, Serial Line Internet Protocol (SLIP)
  - Asynchronous interface: EIA/TIA-232

- ADSL WAN interface card
  - Supports ATM adaptation layer 5 (AAL5) services and applications
  - Interoperates with Alcatel DSL access multiplexer (DSLAM) with Alcatel chipset and Cisco 6130/6260 DSLAM with Globe span chipset
  - Complies with ANSI T1.413 issue 2 and ITU 992.1 (G.DMT)
- G.shdsl WAN interface card
  - Based on the ITU G.991.2, delivers symmetrical data rates from 192 kbps to 2.3 Mbps; speeds vary, depending on loop length and line conditions

- ISDN WAN interface cards
  - ISDN dialup and ISDN DSL (IDSL) at 64 and 128 kbps
  - Encapsulation over IDSL, Frame Relay, and PPP [53], [55], [56], [57]

### 4. 3 Network Transmission Media

In order to carry the data packets from one node to another on a network some sort of media transmission must be employed. There are many variations of such media in

existence, however, only the most widely used types for LANs and WANs connections will be covered.

### 4. 3. 1. Ethernet Cable

In this section, we describe the Ethernet cables that are used to connect the router to the local Ethernet network. A 10/100BASE-TX router, such as the Cisco 1721 router, requires Category 5 Unshielded Twisted-Pair (UTP) or Shielded Twisted-Pair (STP) cable for either straight-through or cross-over Ethernet. Table 4.5 and Table 4.6 illustrate the pin-outs for the straight-through and cross-over Ethernet cable respectively.

Table 4.5: Straight-through Ethernet Cable Pin-outs

| RJ-45 Pin* | Signal | Direction | RJ-45 Pin |
|---|---|---|---|
| 1 | TX+ | —> | 1 |
| 2 | TX- | —> | 2 |
| 3 | RX+ | <— | 3 |
| 6 | RX- | <— | 6 |
| * Pins 4, 5, 7, and 8 are not used for signaling. | | | |

Table 4.6: Cross-over Ethernet Cable Pin-outs

| RJ-45 Pin* | Signal | Direction | RJ-45 Pin |
|---|---|---|---|
| 1 | TX+ | —> | 3 |
| 2 | TX- | —> | 6 |
| 3 | RX+ | <— | 1 |
| 6 | RX- | <— | 2 |
| * Pins 4, 5, 7, and 8 are not used for signaling. | | | |

[53], [54], [55]

### 4. 3. 2 Ethernet Network Cabling Guidelines

In Table 4.7, below, we describe some guidelines for creating cable of Ethernet networks. However, Specs might vary, depending on the manufacturer of the network equipment.

Table 4.7: Ethernet Cabling Guidelines

| Specification | 10BASE-T | 100BASE-TX |
|---|---|---|
| Maximum segment length | 100 meters | 100 meters |
| Maximum number of segments per network | 5 | With Class I repeaters: 1<br>With Class II repeaters: 2 |
| Maximum hop count* | 4 | With Class I repeaters: 0<br>With Class II repeaters: 1 |
| Maximum number of nodes per segment | 1024 | 1024 |
| Cable type required | UTP Cat 3, 4, or 5 | UTP Category 5 or STP |
| * Hop count = Routing metric used to measure the distance between a source and a destination. | | |

### 4. 3. 3 DCE/DTE DB60 Cable

Cisco router to router cable DB60 ends both ends. In other words, this used to connect two Cisco routers via the synchronous serial ports. This cable is a 1ft V.35 DTE/DCE DB60-DB60 Crossover Back-to-Back Cable used to connect Cisco 1600/1700/2500/2600/3600 (WIC-1T, NM-4A/S, NM-8A/S on the 2600 and 3600) Series routers via their serial ports to simulate Frame-Relay and other WAN topologies. Figure 4.5 shows the serial cable sockets that connect between WIC's on two routers.

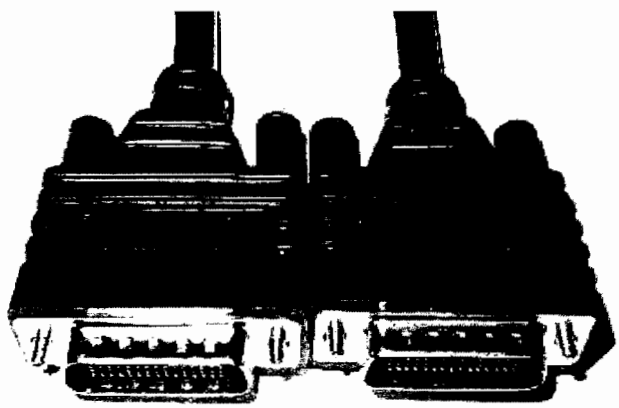

**Figure 4.5: WAN Serial Cable Sockets** [55]

### 4. 3. 4 Console Cable and Adapter

A console cable is provided with the router. We use this cable to connect the router to a PC or terminal. The router comes with a DB-9-to-DB-25 adapter that may be used for connecting the router to a modem, using the console cable. Table 4.8 describes the wiring for the console port and the console cable. This table also includes pin-outs for the DB-9-to-DB-25 adapter.

Table 4.8: Console Cable and Adapter Pin-outs

| **Console (DTE)** | **Console Port** | **Console Cable** | **Adapter** | **Terminal (DTE)** |
|---|---|---|---|---|
| **Signal** | **RJ-45 Pin** | **DB-9 Pin** | **DB-25 Pin** | **Signal** |
| RTS | 1 | 8 | 5 | CTS |
| DTR | 2 | 6 | 6 | DSR |
| TXD | 3 | 2 | 3 | RXD |
| GND | 4 | 5 | 7 | GND |
| GND | 5 | 5 | 7 | GND |
| RXD | 6 | 3 | 2 | TXD |
| DSR | 7 | 4 | 20 | DTR |
| CTS | 8 | 7 | 4 | RTS |

[53], [54], [55], [57]

## 4. 4. The Test-Bed Network Model

The Test-Bed model employed is a combination of hardware and software. The process of model validation is a key for the use of four Cisco Routers, computers, and Java simulation model to evaluate the performance of EIGRP, RIPv2 and OSPF routing protocols in the context of secured and non-secured situations. For the case of providing security constrains within our test-bed model, we illustrate in details the Message Digest (MD5) authentication. This authentication prevents a router from accepting and using unauthorized, malicious, or corrupted routing traffic. Also, we describe a general process that emphasizes three measurements factors for validation: *Average delay, Jitter*, and *overheads*. However, we use a test-bed Java simulation model to generate specific routing traffic data which will pass from client to server specifically in the experiment model. Moreover, a Java-based Object-oriented

discrete-event program with both client and server will be implemented at the end nodes of the network model.

### 4. 4. 1 Message-Digest 5 (MD5) Authentication

General speaking, the damage that can be done in an unsecured routing infrastructure is so enormous that special precautions have to be taken into consideration. Modifying routing tables maliciously can cause significant network traffic to be diverted to the wrong destination. In general, a non-secure routing infrastructure degrades the performance of routers when they are intentionally or unintentionally miss configured. Unfortunately, no widely deployed secure routing protocols are used today. The current way of protecting routing infrastructures relies on so-called *best practices*, which include various simplistic techniques such as firewalls, intrusion detection systems, authentication Message Digest (MD5), route filters, and private addressing [40], [41]. Authentication occurs when any router ensures that only routing updates received from a trusted neighbor are used. This prevents a router from accepting and using unauthorized, malicious, or corrupted routing updates that may compromise the security or availability of the network, and lead, for example, to rerouting of traffic or a denial of service [48[, [58].

### 4. 4. 2. MD5 Algorithm

In cryptography, MD5 is a widely used, partially insecure cryptographic hash function with a 128-bit hash value. As an Internet standard, MD5 has been employed in a wide variety of security applications, and is also commonly used to check the integrity of files. An MD5 hash is typically expressed as a 32 digit hexadecimal number. MD5 was designed by Ron Rivest in 1991 (RFC 1321) to replace an earlier hash function, MD4. In 1996, a flaw was found with the design of MD5. While it was not a clearly fatal weakness, cryptographers began recommending the use of other algorithms, such as SHA-1 (which has since been found vulnerable itself). In 2004, more serious flaws were discovered making further use of the algorithm for security purposes questionable. In 2007 a group of researchers including Arjen Lenstra described how to create a pair of files that share the same MD5 checksum [40], [41], [48].

Figure 4.6 shows one MD5 operation out of 64 operations that are grouped in four rounds of 16 operations. $F$ is a nonlinear function; one function is used in each round. $M_i$ denotes a 32-bit block of the message input, and $K_i$ denotes a 32-bit constant, different for each operation.

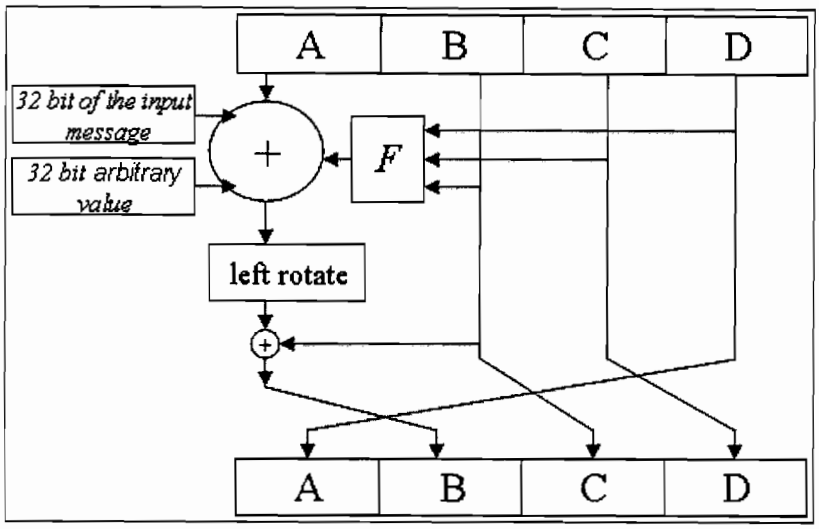


**Figure 4.6: One MD5 operation** [48]

From Figure 4.6, ⊕ denotes addition modulo $2^{32}$. MD5 processes a variable-length message into a fixed-length output of 128 bits. The input message is broken up into chunks of 512-bit blocks (sixteen 32-bit integers); the message is then padded so that its length is divisible by 512. The padding works as follows: first a single bit, 1, is appended to the end of the message. This is followed by as many zeros as are required to bring the length of the message up to 64 bits less than a multiple of 512. The remaining bits are filled up with a 64-bit integer representing the length of the original message, in bits. The main MD5 algorithm operates on a 128-bit state, divided into four 32-bit words, denoted $A$, $B$, $C$ and $D$. These are initialized to certain fixed constants. The main algorithm then operates on each 512-bit message block in turn, each block modifying the state. The processing of a message block consists of four similar stages, termed *rounds*; each round is composed of 16 similar operations based on a non-linear function $F$, modular addition, and left rotation [40], [41], [48].

### 4. 4. 3. MD5 Applications

MD5 digests have been widely used in the software world to provide some assurance that a transferred file has arrived intact. For example, file servers often provide a pre-

computed MD5 checksum[1] for the files, so that a user can compare the checksum of the downloaded file to it. Unix-based operating systems include MD5 sum utilities in their distribution packages, whereas Windows users use third-party applications. However, now that it is easy to generate MD5 collisions, it is possible for the person who created the file to create a second file with the same checksum, so this technique cannot protect against some forms of malicious tampering. Also, in some cases the checksum cannot be trusted (for example, if it was obtained over the same channel as the downloaded file), in which case MD5 can only provide error-checking functionality through recognizing a corrupt or incomplete download, which becomes more likely when downloading larger files. MD5 is widely used to store passwords. To do against the vulnerabilities mentioned above, one can add a salt, comprises random bits that are used as one of the inputs to a key derivation function, to the passwords before hashing them. Some implementations, however, may apply the hashing function even more than once [40], [41], [48].

### 4. 4. 4. MD5 hashes

The MD5 hash known as checksum for a file is a 128-bit value, something like a fingerprint of the file. There is a very small possibility of getting two identical hashes of two different files. This feature can be useful both for comparing the files and their integrity control. The 128-bit (16-byte) MD5 hashes, also termed message digests, are typically represented as a sequence of 32 hexadecimal digits. For example, the following 43-byte ASCII input message ("The quick brown fox jumps over the lazy dog") corresponds to the MD5 hash 9e107d9d372bb6826bd81d3542a419d6. However, even a small change in the message, with overwhelming probability, will result in a completely different hash, due to the avalanche effect. For instance, adding a period to the end of the above message would result in producing the following MD5 hash: e4d909c290d0fb1ca068ffaddf22cbd0 [48], [62].

In MD5 authentication, the participating routers must share an authentication key. This key must be manually preconfigured on each router. For EIGRP, multiple keys

[1] * A **checksum** is a form of redundancy check, a simple way to protect the integrity of data by detecting errors in data that are sent through space (telecommunications) or stored for some time

can be used for authentication. Each key is associated with a number, which must be the same for all the routers and never be sent over the wire. Each router uses a combination of this number and the traffic data as inputs to the MD5 algorithm to produce a message digest called hash. For RIPv2, when keyed MD5 is used, the same header and content are used, except that the 16-byte authentication key field is reused to describe a Keyed Message Digest trailer. For OSPF, the OSPF packet header includes an Authentication Type field and 64 bits of data for use by the appropriate authentication scheme. Generally, most fields within this common header have obvious meanings. For instance, the version number is set to 2 to indicate OSPFv2 and the type is the OSPF packet type i.e. hello, database description, link-state request, link-state update, and link-state acknowledgment. The packet length is the number of bytes in the packet. Figure 4.7 illustrates the sequence of events involved in MD5 authentication for the sending router [40], [41], [48].

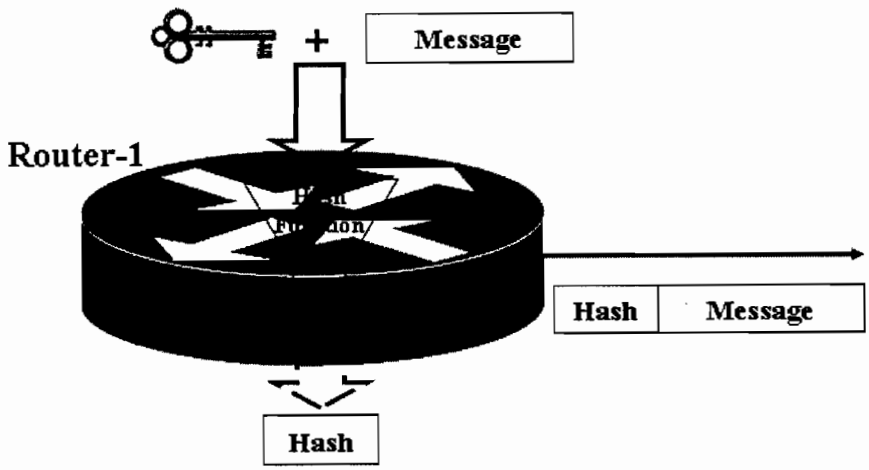


**Figure 4.7: MD5 Neighbor Authentication at the Sender Router**

Accordingly, the MD5 algorithm takes the preconfigured shared secret key and the traffic data or message as inputs and returns a message digest, hash, which is appended to the message and sent through the appropriate interface. Figure 4.8 illustrates the sequence of events for routing protocol authentication at the destination router [14].

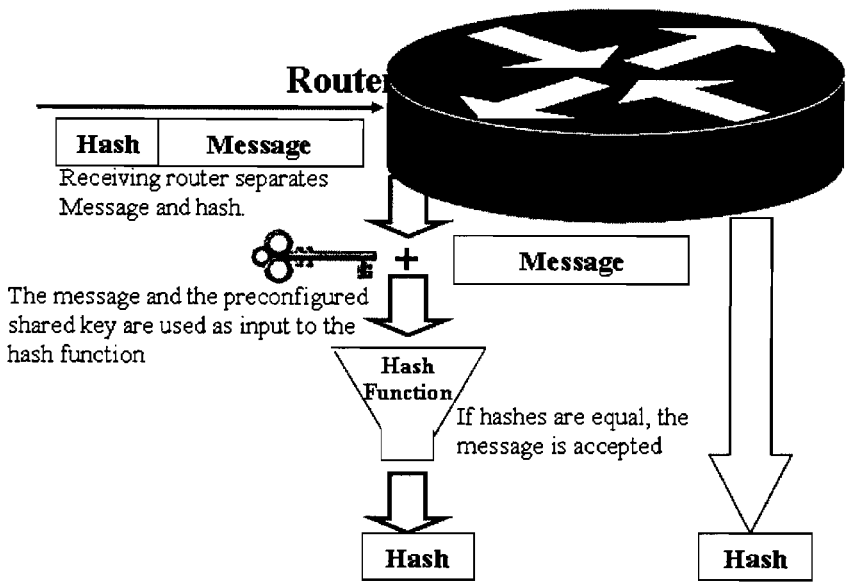


**Figure 4.8: The Sequence of Events at the Destination Router** [14]

EIGRP, RIP and OSPF are supported keyed MD5 cryptographic checksums to provide authentication of traffic data including routing updates. Each key is represented by key number, key string, and key identifier, which are stored locally. EIGRP MD5 authentication supports multiple keys, which are grouped in one keychain. RIP MD5 the basic RIPv2 message format provides for an 8-byte header with an array of 20-byte records as its data content. When keyed MD5 is used, the same header and content are used, except that the 16-byte authentication key field is reused to describe a Keyed Message Digest trailer. With MD5, all OSPF protocol exchanges are authenticated. The OSPF packet header includes an Authentication Type field and 64 bits of data for use by the appropriate authentication scheme. Each key has a lifetime period that validates the usage of this key for sending and receiving. The router selects one key from the keychain for sending an authentication packet. The key numbers are examined from the lowest to the highest, and the first valid key encountered is used [14], [41], [48].

### 4. 4. 5. End-to-End Client /Server

Client/server networking grew in popularity many years ago as personal computers (PCs) became the common alternative to older *mainframe* computers. Client devices are typically PCs with windows XP operating system and network software

applications installed that request and receive information over the network. The client initiates requests from the server, waits for replies and then receives replies, it usually connects to a small number of servers at one time, also, interacts directly with end-users using a graphical user interface.

A server is device typically stores files and databases including more complex applications like Web sites. Server devices often feature higher-powered central processors, more memory, and larger disk drives than clients. The server never initiates requests or activities; it listens to network and responds only to requests from connected and authorized clients, and waits for and replies to requests from connected clients [59], [60]. Table 4.9 illustrates the client and server personal computer specifications. This client/server is Windows platform.

**Table 4.9.: Client-Server Personal Computer Specifications**

| Client | Server |
|---|---|
| Personal computer Dell, Latitude D-610 Laptop, with the following specs:<br>- Processor: CPU 2.0 GHz.<br>- RAM: 512 Giga Bytes.<br>- Hard Disk: 80 Giga Bytes.<br>- Operating System: Windows XP professional with Service Pack 2.<br>- Integrated Network Card 10/100 MB. | Personal computer Dell, OptiPlex GX 260, with the following configurations:<br>- Processor: CPU 2.8 GHz.<br>- RAM: 1.5 Giga Bytes.<br>- Hard Disk: 80 Giga Bytes.<br>- Operating System: Windows XP professional with Service Pack 2.<br>- Integrated Network Card 10/100 MB. |

### 4. 4. 6. The Client/Server Simulation

Generally speaking, the Java language environment, and client-server architecture are complementary software technologies, which, when used together provide a powerful set of tools for developing and deploying generate, send, and receive number of specific packets.

The idea of adding process-oriented simulation capabilities to a general purpose object-oriented programming language is not new. The Java language has several features that are ideally suited to the implementation of advanced discrete-event simulation architectures and reusable simulation software components. One is a

simple yet powerful framework that greatly facilitates the implementation of object-oriented design methodology and its capabilities for creating flexible, modular, and reusable programs. Also, where other languages rely on a host of third-party supported libraries, Java includes native support for networking and common Internet protocols, database connection via Java Database Connectivity, multithreading, distributed objects via Remote Method Invocation, and graphical user interfaces via the Abstract Windowing Toolkit. [56], [57], [58].

In this research, a Java-based Object-oriented discrete-event program with both client and server is implemented at the end nodes of the network model. The network traffic, namely TCP packets, is directed from the client to the server, which calculates the major performance measures, especially the average delay time of the TCP packets. The packet data size is set to 1000 bytes and the generation of these packets follows the Markov Poisson Process (MPP), which is a doubly stochastic Poisson process whose rate varies according to a Markov process. The MPP can be viewed as a superposition of latent Poisson processes, which can be expressed as a non-homogeneous discretely indexed Hidden Markov Model (HMM) by partitioning time into intervals between observed events. The resultant traffic model is an ON/OFF traffic where the client sends bulk traffic during the ON periods and nothing during the OFF periods. ON and OFF periods are distributed exponentially with a mean of 10. The number of packets in bulk traffic is distributed normally with mean equals to 100 and variance equals to 10 [40], [41].

We describe an approach to build easy-to-use client software named *Client* Java program, which use to generate the packets and pass it to a remote server. On the other hand, we build server software named *Server* Java program, which received these packets and calculate the average delay time, jitter and overhead for number of bulk packets. We use this scheme for evaluating and developing a generic framework for processing network packets.

The synchronization issue in the client/server environment should be highly considered. To do so, we use a Clock synchronous v1.0.0, Miro-Karjalainen [63] program to synchronize the time between the client and the server to be in the same time which is measured by milliseconds.

### 4. 4. 7. The Client Side

The Client generates a specific number of packets that will be transmitted to the server on the other side through the test-bed network model which has been employed for this research. Figure 4.9 illustrates the logic of the Client which starts with setting the packet size to 1000 bytes. This is equal to 8000 bits as each byte holds 8 bits. The Client logic is composed of three main parts. The first part is concern with configuring a selected traffic pattern while the second part establishes connection with the server and lastly, the third part executes the selected traffic pattern, i.e. transmits packets to the server following the chosen traffic pattern. In the first part, the selected traffic pattern is MPP for which OFF period lengths are exponentially distributed with a given mean (OFFGen: *exponential*(*mean*)). The lengths of the ON periods are dependent on the size of bulks to be transmitted during these periods. The size (*BSize*) of a bulk is deterministic starting at size of 10,000 packets for the first bulk and increasing by a fixed increment of 5,000 packets for the next bulk. Targeted maximum bulk size is set to 55,000 packets. In the second part a socket connection is established with the server on $IP_x$ and $Port_y$. Finally, the third part implements the selected and configured traffic pattern of part one and streaming the packets to the server over the network.

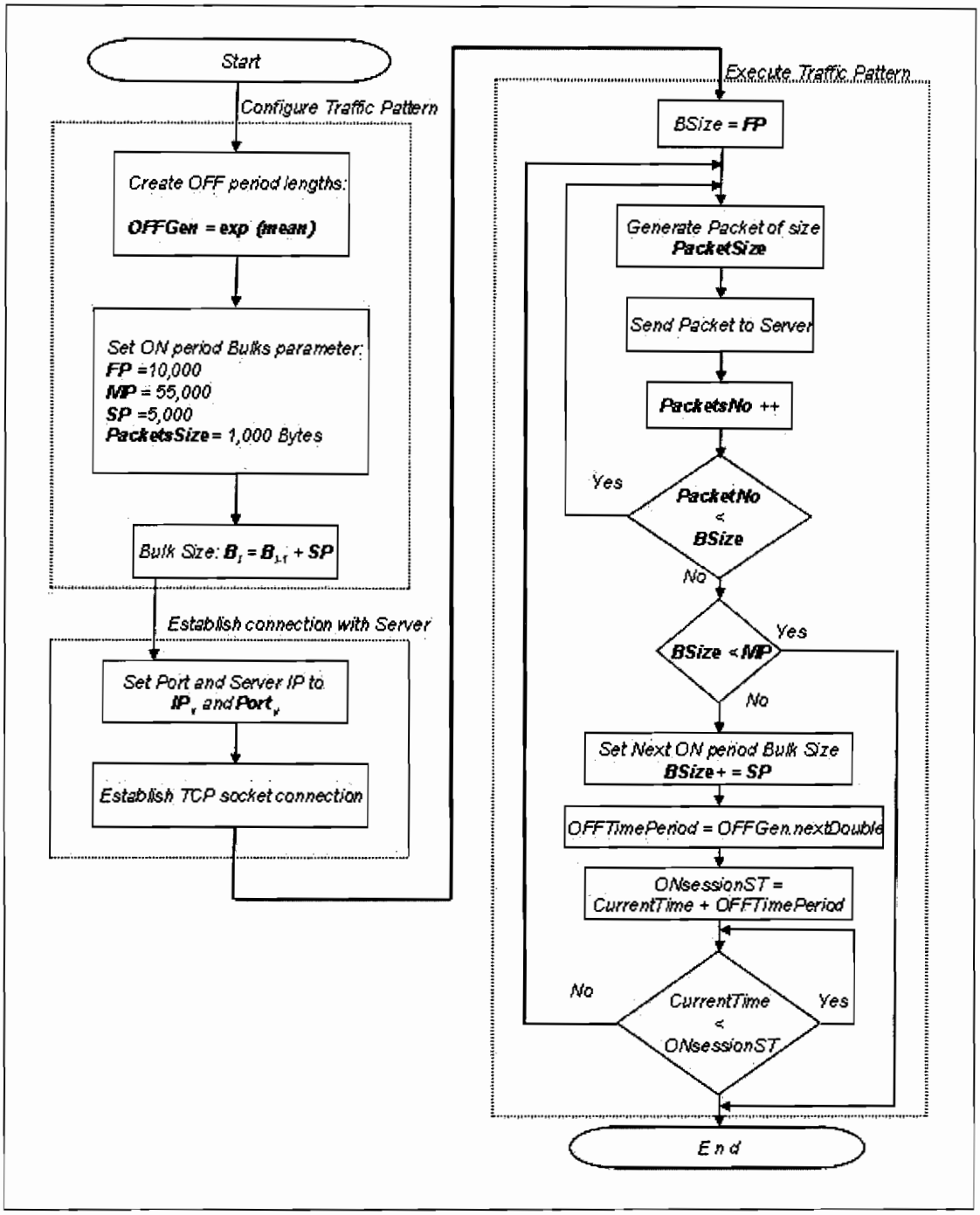


**Figure 4.9: Client Logic**

We are using a JCreator LE 4.0 to run the Client Java simulation program. Figure 4.10 shows snapshoot of program compilation where the process is correctly completed.

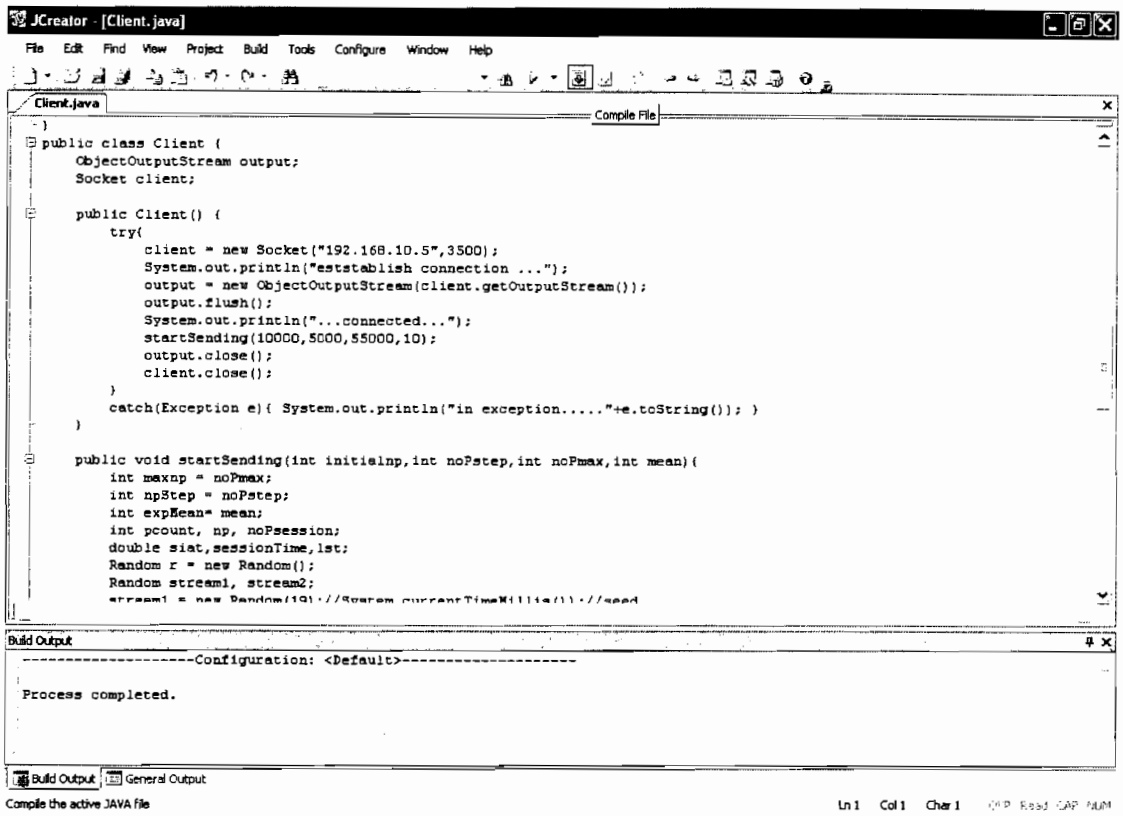


**Figure 4.10: Client Java program after compilation**

### 4. 4. 8. The Server Side

The Server receives bulks of packets, sent from the Client through the test-bed network model which has been deployed for this research, calculates the average delay time, jitter and overhead for these bulks of packets and eventually outputs the results in a text file. Accordingly as Figure 4.11 illustrates, to start receiving the client's bulks of packets the Server establishes its server socket on $IP_x$ and $Port_y$. This can be done through forcing the server socket to listen for Client connections in an infinite loop. Once a Client request is received, the Server creates a socket to deal with the requesting client then starts a process to receive the client's bulks of packets. Whenever packets are received, they are processed and their relevant performance measures of interest are calculated; i.e. average delay, jitter and overhead. The collected results will then be saved in a separate text file. Similarly, the Server keeps in checking whether the Maximum number of packets (*MP*) counter has reached its limit of 55,000 packets before getting the new First number of packets (*FP*) value, equals to the previous FP value plus the SP value, in the next cycle of receiving and

so on. Eventually, when MP counter exceeds the value of 55,000 packets the Server simulation program will stop. In addition, as shown in Figure 4.12, the server has to get ready waiting for the Client connection request.

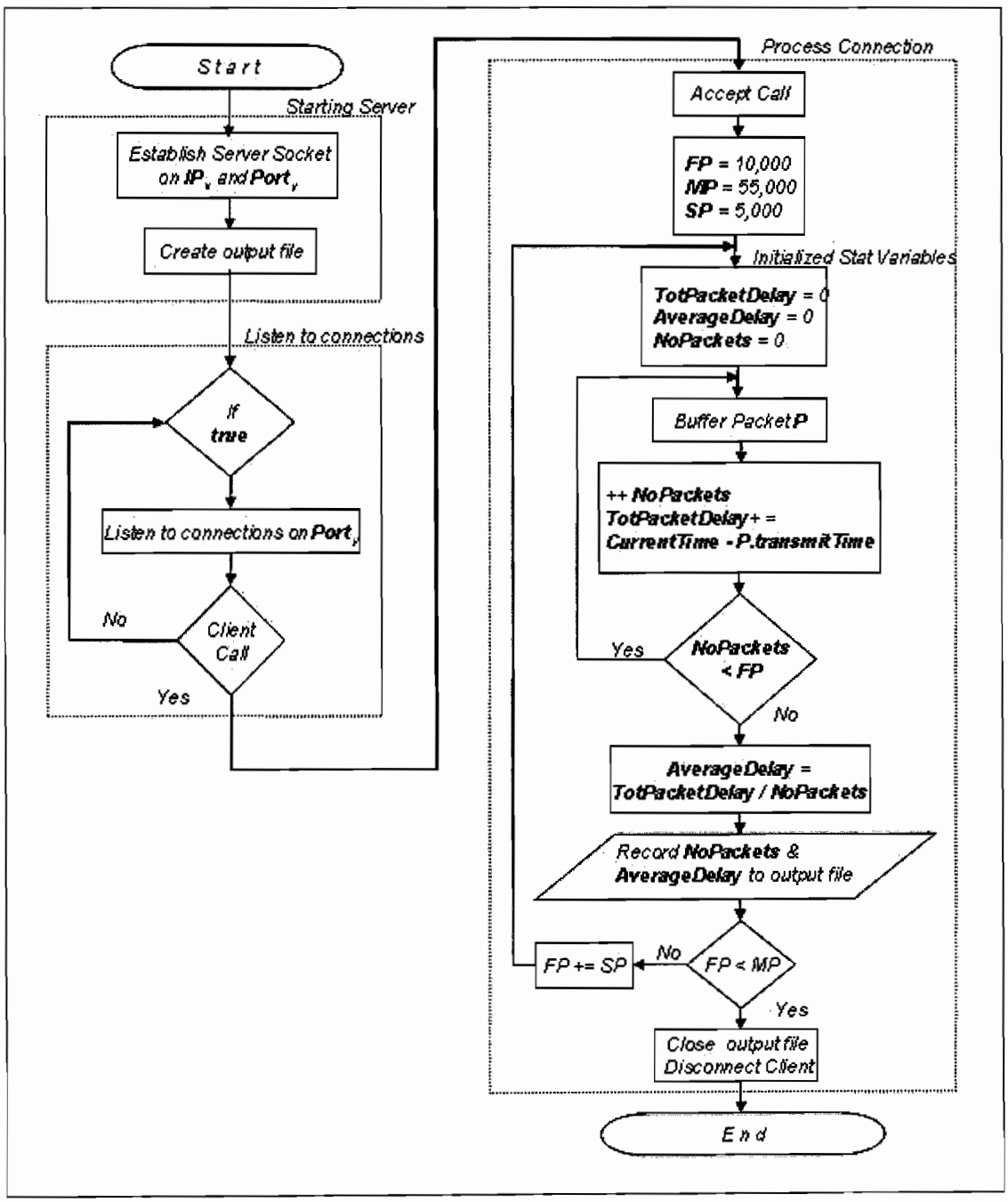


**Figure 4.11: Server Logic**

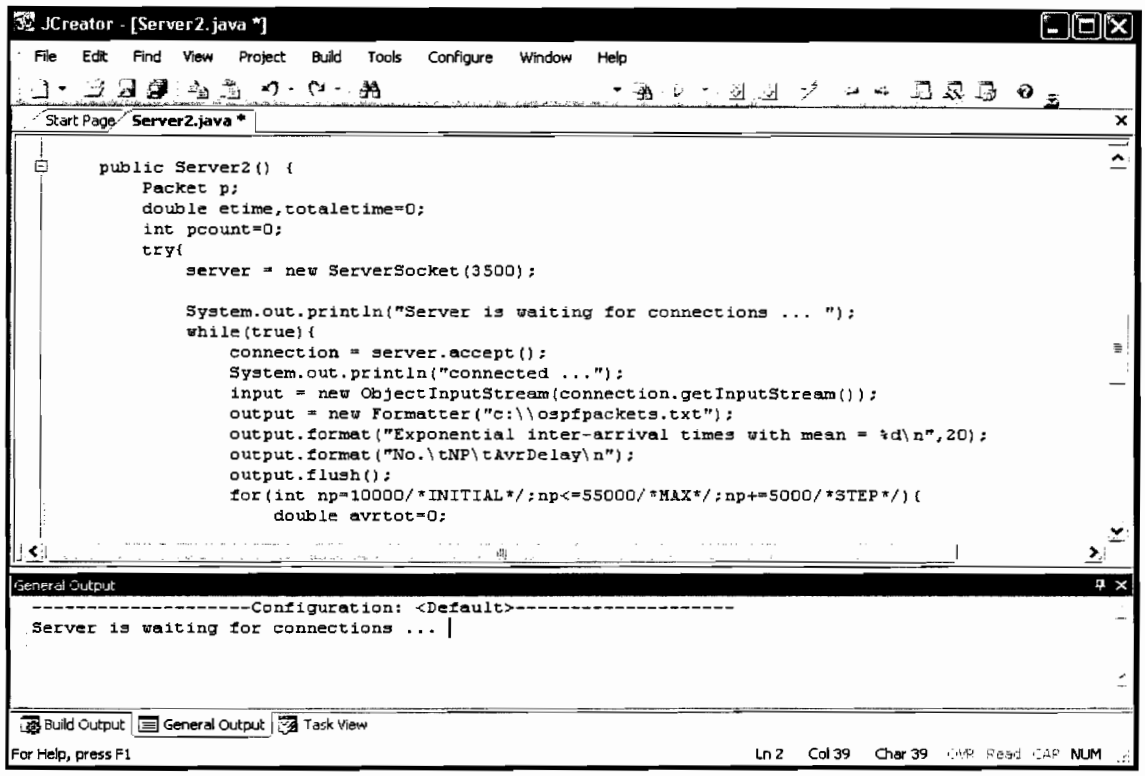


**Figure 4.12: Server Waiting for Connections**

Again, we are using a JCreator LE 4.0 to run the Server Java simulation program. Figure 4.13 shows snapshoot of program compilation with the process is correctly completed.

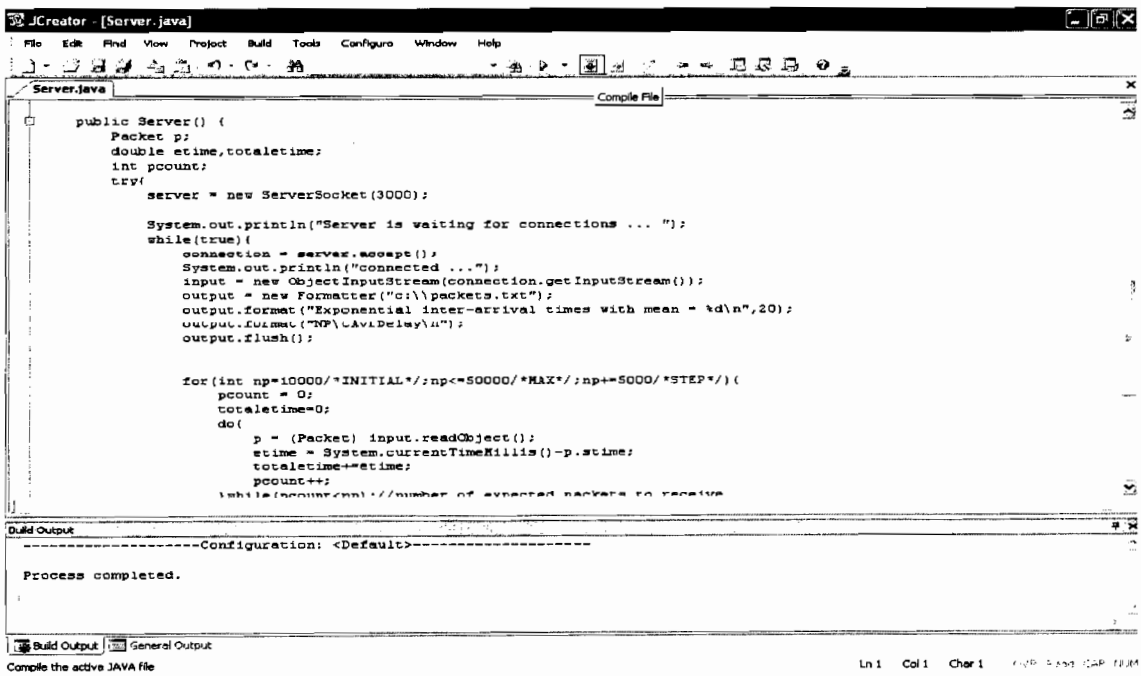


**Figure 4.13: Server Java program after compilation**

### 4.4.9. Model Traffic Pseudo Code

In this section, we show our five-step model traffic pseudo code as illustrated in Figure 4.14. At step zero, we setup our network model as a collection of clients, routers, and single server. This has clearly illustrated earlier in Figure 4.1. Next at step one; we represent our synchronization model components which include the operations of Synchronizing routers physical clocks and Synchronizing Server, Clients and Routers logical clocks. On step two, we define our adopted routing protocols as EIGRP, RIPv2, and OSPF. We also describe our MD5 authentication algorithm which ultimately will be used to enforce our desired routing traffic security constrains. At step three, we select and configure the traffic pattern, establish server connections at the specified network address and port, serve the packets, calculate their performance measures of interests, and finally stores the collected results in an output file. Eventually at step five, we do the comparative analysis for the three routing protocols.

```
0: Setup network model: Collection of clients, routers and a Server from the sets C=
        {c1, c2, ... , cn}, R={ r1,r2,..rm}, and S= {s1} respectively.
1: Synchronization model components
        1:1 Synchronize routers physical clocks
        1:2 Synchronize Server, Clients and Routers logical clocks
2:  Set D= P×O, where P= {EIGRP, RIPv2, OSPF} and O= {non-Secured, MD5
        Secured}
3: Loop: ∀d∈D
            - Loop ∀rj∈R, where j∈{1,2,...,m}:
                - Setup & configure d on rj
            - End loop
            - Start Server with IPx, Porty
            - Loop: ∀ci∈C, where i∈{1,2,...,n}
                - Select & Configure a traffic pattern
                - Loop: iteration≤ max_iterations
                    - ∀ci∈C, where i∈{1,2,...,n} establish connections to s1
                      at IPx, Porty
                    - Loop: No_packets ≤ max_packets
                        - Simultaneously:
                            - Arbitrary ∀ci∈C, where i∈{1,2,...,n}
                              plug traffic to N
                            - s1:
                                - processes packets
                                - Calculates measures
                                - Saves measures to output file
                    - End loop
                    - Calculate overall weighted measure for this iteration
                    - ∀ci∈C, where i∈{1,2,...,n} disconnect from s1
                - End Loop
                - Calculate overall weighted measure ∀iterations.
            - End Loop
        - End Loop
4: Compare All results ∀d∈D.
```

**Figure 4.14: Five-Step Model Traffic Pseudo Code**

### 4. 4. 10. Traffic Pattern Model

Generally speaking, the last few years have seen a rapid growth in both the volume and variety of network traffic, while at the same time it is becoming ever more important to understand network behaviors to provide security and misuse monitoring.

To model network traffic, the system is initialized with a set of interests, such as my idea that shown in figure 4.15. The interests describe a variety of traffic characteristics: some about the source or destination, the type of connection, the characteristics of the traffic, temporal relations and trends, variability etc. [4.12]

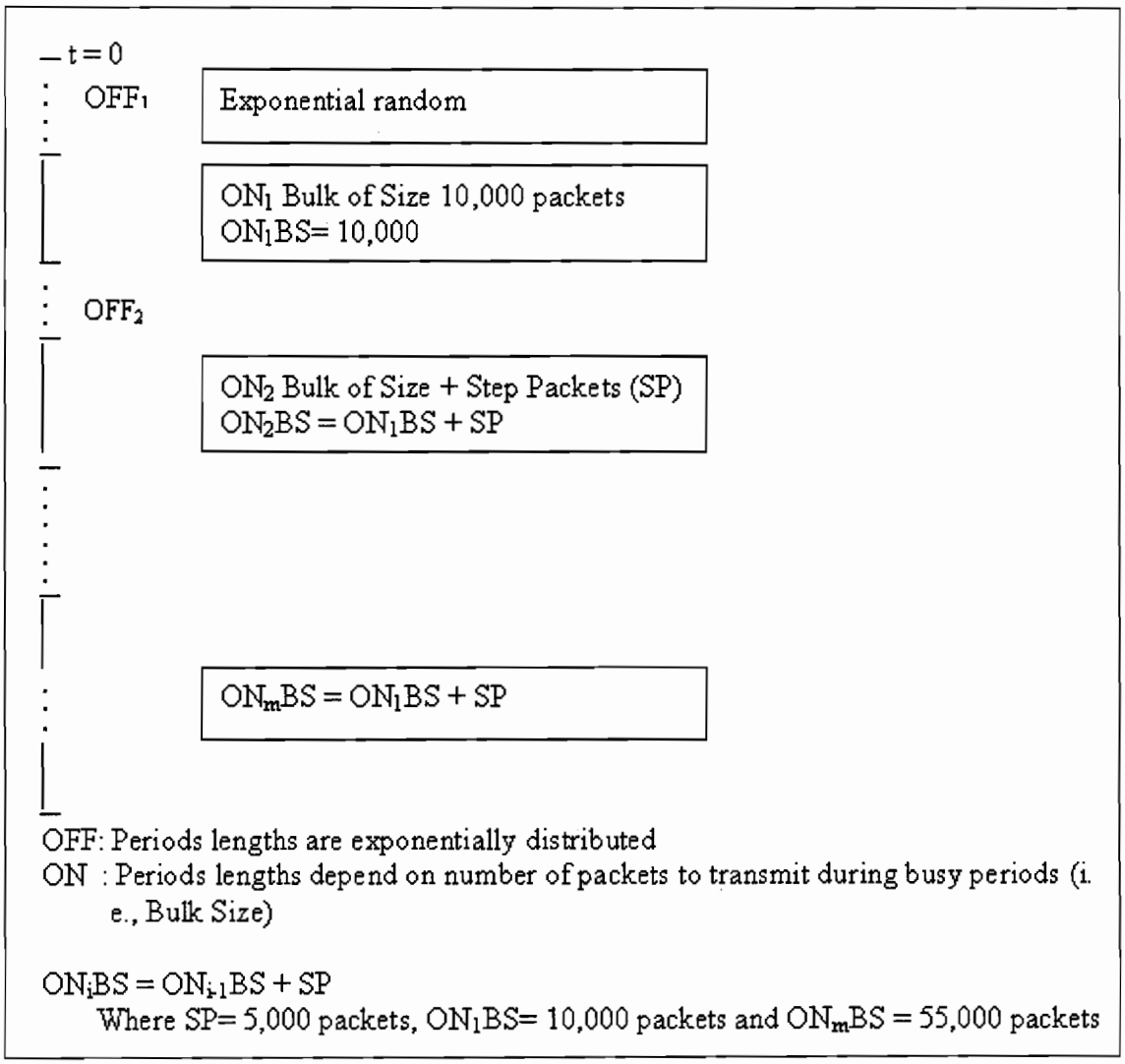


**Figure 4.15: Model Traffic Pattern**

### 4. 4. 11. Routing Protocols Configurations

Generally, most people configure their routers by using *telnet* or windows Hyper-terminal. Initially, we turn on the router and connect its console port to the PC/terminal serial (COM) port using a rollover, named console cable. As in Figure 4.16, we start a windows Hyper-terminal session and set COM1 port to be used and then click OK. Eventually, set the speed of the connection to 9600 baud rate and click OK as shown in Figure 4.17.

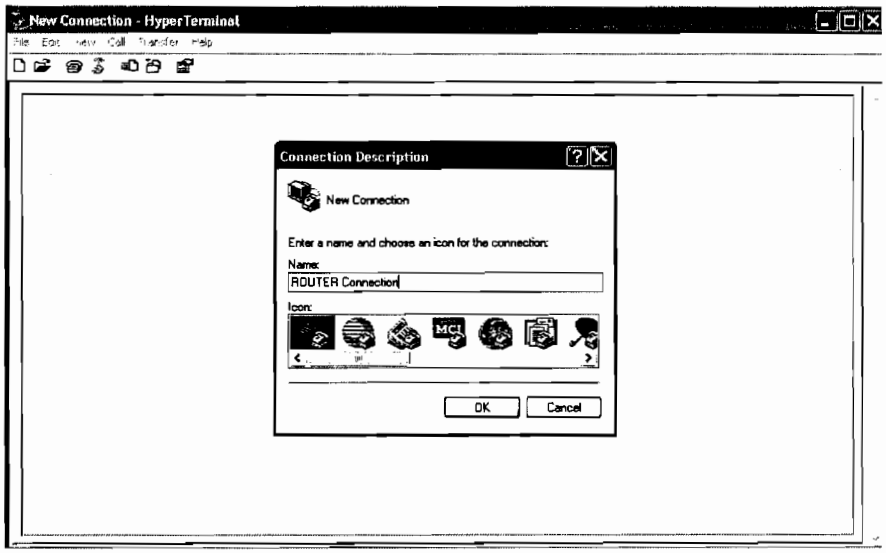


**Figure 4.16: Start Hyper-terminal Connection**

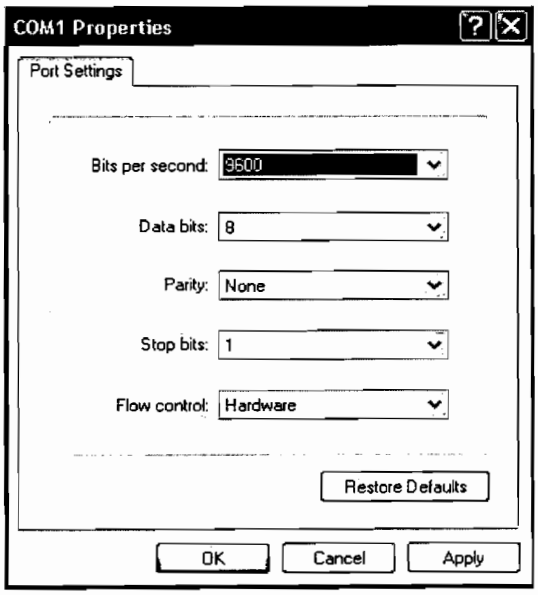


**Figure 4.17: Setting up Hyper-terminal Connection**

Now when the router boots up, it asks if you wish to begin by using the initial configuration. Hit the Enter key to see the prompt from the router as it will look like this:

ROUTER1>

Issue the command *enable* and provide the password to enter privileged mode.

ROUTER1> enable

This is the privilege mode with a pound sign (#):

ROUTER1#

Once we enter privilege mode, we have access to all the configuration information and options the IOS provides, either directly from privilege mode, or from one of its sub-modes. Each of these modes has a prompt of the form:

To configure any feature of the router, we must enter configuration mode.

ROUTER1# configure terminal

ROUTER1(config)#interface serial 0

ROUTER1(config-if)#

Or

ROUTER1(config)#router eigrp 100

ROUTER1(config-router)#

As demonstrated above, the prompt change indicates the mode that you are working in.

In fact, the router's configuration file has two important configuration files. There is the configuration file that describes the current running state of the router, which is called the *running-config*. Also, there is a configuration file that the router uses to boot, namely called the *startup-config*. However, for our research, we mainly focus on the following parts:

- Configuring interfaces.
- Configuring EIGRP, RIPv2, and OSPF routing protocols.
- Configuring MD5 authentication in secured mode.

Figure 4.18 illustrates a sample EIGRP configuration carried out on all test-bed model routers that are employed in this research for the case of secured MD5 Authentication.

**Configure EIGRP to perform Secured MD5 Authentication mode:**

*Enter global configuration mode:*

ROUTER1**#configure terminal**

*From global configuration mode, specify the interface that you want to configure EIGRP message authentication on. This on all Interfaces, the sample here is on fastethernet 0:*

ROUTER1 (config)#**interface fastethernet 0**

ROUTER1 (config-if)#**ip address 192.168.1.1 255.255.255.0**

*From global configuration mode, specify the clockrate that you want to configure.*

ROUTER1 (config-if)#**clockrate 800000** (on DCE cable)

ROUTER1 (config–if)#**ip authentication mode eigrp 100 md5**

*Specify the keychain that should be used for authentication*

ROUTER1 (config–if)#**ip authentication key–chain eigrp 100 khalidchain**

*We did the same configuration on other interfaces* ***Serial 0 & Serial 1***

ROUTER1 (config–if)#**no shutdown**

ROUTER1 (config–if)#**end**

*Create the key chain*

ROUTER1 (config)#**key chain khalidchain**

*Specify the key number*

ROUTER1 (config–keychain)#**key 1**

*Specify the key–string for the key*

ROUTER1 (config–keychain–key)#**key–string Khalid-63**

We configure key management commands in key-chain key configuration mode to automatically migrate from one authentication key to another by configuring the:

***accept-lifetime*** *start-time {infinite | end-time | duration seconds}*

***send-lifetime*** *start-time {infinite | end-time | duration seconds}*

ROUTER1(config–keychain–key)#**accept-lifetime 00:00:00 May 31 2008 infinite**

ROUTER1(config–keychain–key)#**send-lifetime 00:00:00 May 31 2008 infinite**

*End the configuration mode*

ROUTER1 (config–keychain–key)#**end**

*Enable EIGRP message authentication. The 100 used here is the autonomous system number of the network. MD5 indicates that the MD5 hash is to be used for authentication*

We then configure EIGRP to perform MD5 authentication using the key as shown:

*Enter global configuration mode*

ROUTER1#**configure terminal**

ROUTER1(config)#**router eigrp 100**

ROUTER1(config-router)# **network 192.168.1.0**

ROUTER1(config-router)# **network 192.168.101.0**

ROUTER1(config-router)# **network 192.168.102.0**

ROUTER1 (config–router)#**end**

**Figure 4.18: EIGRP Configuration in Secured MD5 Authentication**

Figure 4.19 illustrates a sample EIGRP configuration carried out on all test-bed model routers that are employed in this research for the case of non-secured MD5 Authentication.

**Configure EIGRP to perform a non-secured mode:**

*Enter global configuration mode*

ROUTER1#**configure terminal**

*From global configuration mode, specify the interface(s) that you want to configure, the sample here is on fastethernet 0:*

ROUTER1 (config)#**interface fastethernet 0**

ROUTER1 (config-if)#**ip address 192.168.1.1 255.255.255.0**

*From global configuration mode, specify the clockrate that you want to configure.*

ROUTER1 (config-if)#**clockrate 800000** (on DCE cable)

ROUTER1 (config–if)#**no shutdown**

ROUTER1(config–if)#**end**

*From global configuration mode enable EIGRP as routing protocol. The 100 used here is the autonomous system number of the network.*

ROUTER1(config)#**router eigrp 100**

*Specify the network numbers that will be routing between themselves that should be*

*used.*

ROUTER1#**configure terminal**

ROUTER1(config)#**router eigrp 100**

ROUTER1(config-router)# **network 192.168.1.0**

ROUTER1(config-router)# **network 192.168.101.0**

ROUTER1(config-router)# **network 192.168.102.0**

ROUTER1(config–router)#**end**

**Figure 4.19: Non-Secured EIGRP Configuration**

Figure 4.20 illustrates a sample RIPv2 configuration carried out on all test-bed model routers that are employed in this research for the case of secured MD5 Authentication.

**Configure RIPv2 to perform secured MD5 Authentication mode:**

*Enter global configuration mode:*

ROUTER1**#configure terminal**

*From global configuration mode, specify the interface that you want to configure RIPv2 message authentication on all interfaces. The sample here is on fast 0:*

ROUTER1 (config)#**interface(s)**

ROUTER1 (config-if)#**ip address 192.168.5.1 255.255.255.0**

*From global configuration mode, specify the clockrate that you want to configure.*

ROUTER1 (config–if)#**clockrate 800000** (on DCE cable)

ROUTER1 (config–if)# **ip rip authentication mode md5**

*Specify the keychain that should be used for authentication*

ROUTER1 (config–if)# **ip rip authentication key-chain khalidchain**

ROUTER1 (config–if)#**no shutdown**

ROUTER1 (config–if)#**end**

*Create the key chain*

ROUTER1 (config)#**key chain khalidchain**

*Specify the key number*

ROUTER1 (config–keychain)#**key 1**

*Specify the key–string for the key*

ROUTER1 (config–keychain–key)#**key–string khalid-63**

We configure key management commands in key-chain key configuration mode to automatically migrate from one authentication key to another by configuring the ***accept-lifetime*** *start-time {**infinite** | end-time | **duration** seconds} and* ***send-lifetime*** *start-time {**infinite** | end-time | **duration** seconds} (optional)*

ROUTER1 (config–keychain–key)#**accept-lifetime 00:00:00 May 31 2008 infinite**

ROUTER1 (config–keychain–key)#**send-lifetime 00:00:00 May 31 2008 infinite**

*End the configuration*
ROUTER1 (config–keychain–key)#**end**
*Enable RIPv2 message authentication. MD5 indicates that the MD5 hash is to be used for authentication*
We then configure RIPv2 to perform MD5 authentication using the key as shown:
*Enter global configuration mode*
ROUTER1#**configure terminal**
ROUTER1(config)#**router rip**
ROUTER1(config-router)#**version 2**
ROUTER1(config-router)# **network 192.168.1.0**
ROUTER1(config-router)# **network 192.168.101.0**
ROUTER1(config-router)# **network 192.168.102.0**
ROUTER1(config–router)#**end**

**Figure 4.20: RIPv2 Configuration in Secured MD5 Authentication**

Figure 4.21 illustrates a sample RIPv2 configuration carried out on all test-bed model routers that are employed in this research for the case of non-secured MD5 Authentication.

**Configure RIPv2 to perform an non-secured mode:**
*Enter global configuration mode*
ROUTER1#**configure terminal**
*From global configuration mode, specify the interface(s) that you want to configure. The sample here is on fastethernet 0:*
ROUTER1 (config)#**interface(s)**
ROUTER1 (config-if)#**ip address 192.168.5.1 255.255.255.0**
*From global configuration mode, specify the clockrate that you want to configure.*
ROUTER1 (config–if)#**clockrate 800000** (on DCE cable)
ROUTER1 (config–if)#**no shutdown**
ROUTER1 (config–if)#**end**
*From global configuration mode enable RIPv2 as routing protocol. v2 is the version of routing protocol.*
ROUTER1 (config)#**router rip**
ROUTER1 (config-router)#**version 2**
*Specify all the network numbers that will be routing between themselves that should*

*be used.*

ROUTER1(config-router)# **network 192.168.1.0**

ROUTER1(config-router)# **network 192.168.101.0**

ROUTER1(config-router)# **network 192.168.102.0**

ROUTER1 (config–router)#**end**

**Figure 4.21: Non-Secured RIPv2 Configuration**

Figure 4.22 illustrates a sample OSPF configuration carried out on all test-bed model routers that are employed in this research for the case of secured MD5 Authentication.

**Configure OSPF to perform secured MD5 Authentication mode:**

*Enter global configuration mode:*

ROUTER1**#configure terminal**

*From global configuration mode, specify the interfaces that you want to configure OSPF message authentication on. The sample here is on fastethernet 0:*

ROUTER1 (config)#**interface fastethernet 0**

ROUTER1 (config-if)#**ip address 192.168.1.1 255.255.255.0**

ROUTER1 (config–if)# **ip ospf message-digest-key 1 md5 Khalid-63**

*From global configuration mode, specify the clockrate that you want to configure.*

ROUTER1 (config–if)#**clockrate 800000** (on DCE cable)

ROUTER1 (config–if)#**no shutdown**

ROUTER1 (config–if)#**end**

*Enable OSPF message authentication. MD5 indicates that the md5 hash is to be used for authentication. The 100 used here is the autonomous system number of the network.*

We then configure OSPF to perform MD5 authentication using the key as shown:

*Enter global configuration mode*

ROUTER1(config)#**router ospf 100**

ROUTER1(config-router)#**area 0 authentication message-digest**

ROUTER1(config-router)#**network 192.168.1.0 0.0.0.255 area 0**

ROUTER1(config-router)#**network 192.168.101.0 0.0.0.255 area 0**

ROUTER1(config-router)#**network 192.168.102.0 0.0.0.255 area 0**

ROUTER1 (config–router)#**end**

**Figure 4.22: OSPF Configuration in Secured MD5 Authentication**

Figure 4.23 illustrates a sample OSPF configuration carried out on all test-bed model routers that are employed in this research for the case of non-secured MD5 Authentication.

**Configure OSPF to perform a non-secured mode:**

*Enter global configuration mode*

ROUTER1#**configure terminal**

*From global configuration mode, specify the interface(s) that you want to configure. The sample here is on fastethernet 0:*

ROUTER1 (config)#**interface fastethernet 0**

ROUTER1 (config–if)# **ip address 192.168.1.1 255.255.255.0**

ROUTER1 (config–if)#**no shutdown**

*From global configuration mode, specify the clockrate that you want to configure.*

ROUTER1 (config–if)#**Clockrate 800000** (on DCE cable)

ROUTER1 (config–if)#**end**

*From global configuration mode enable RIPv2 as routing protocol. v2 is the version of routing protocol. The 100 used here is the autonomous system number of the network.*

ROUTER1 (config)#**router ospf 100**

*Specify all the network numbers that will be routing between themselves that should be used.*

ROUTER1(config)#**router ospf 100**

ROUTER1(config-router)#**network 192.168.1.0 0.0.0.255 area 0**

ROUTER1(config-router)#**network 192.168.101.0 0.0.0.255 area 0**

ROUTER1(config-router)#**network 192.168.102.0 0.0.0.255 area 0**

ROUTER1(config–router)#**end**

**Figure 4.23: Non-Secured OSPF Configuration**

## 4.5 Conclusion

In this chapter, we explained in details our experiment from both hardware and software perspectives. A Java client and server programs for generating, monitoring traffic and reporting results was presented as part of this work. In

addition, we described the detailed routers' configuration for the purpose of studying the impact of secured MD5 authentication versus non-secured on routing traffic for the cases of EIGRP, RIPv2, and OSPF routing protocols.

# Chapter 5

## Results and Discussion

### 5.1 Introduction

In this chapter, we will evaluate our experimental proposed test-bed network model. The evaluation has three objectives. First, is to show that the average delay time for the secured link state OSPF routing protocol is always less than the secured distance vector RIPv2 and EIGRP routing protocols. Second, is to show that the measuring of jitter for the different routing protocols, which is a variance in delay in individual data packets within a data packet stream, can be visible through variations in amplitude, signal strength, and other elements of such waves. The usual causes include connection timeouts, connection time delay, data traffic congestion, and interference. Third, and finally, is to measure the overhead for the same EIGRP, RIPv2 and OSPF routing protocols.

Following next, fourteen graphs are plotted to evaluate the average delay time, jitter and overhead with respect to the number of packets. Various traffic loads described by total number of packets sent during the sessions of the ON periods have been plugged into the simulation model. Initially a total of 10,000 packets as a first traffic load incremented by 5,000 packets up to 55,000 packets have been used. The following figures show the average delay time, jitter and overhead with number of packets in the case of secured MD5 authentication and unsecured of EIGRP, RIPv2, and OSPF routing protocols.

### 5.2 Enhanced Interior Gateway Routing Protocol (EIGRP)

In the following subsections, three graphs were plotted to evaluate the efficiency of the EIGRP in both non-secured and secured MD5 authentication with respect to the average delay, jitter, and overhead respectively.

### 5.2.1 Average Delay Time of EIGRP in both Secured/Unsecured

Figure 5.1 shows the average delay time with number of packets in the secured MD5 authentication case and unsecured case for EIGRP routing protocol.

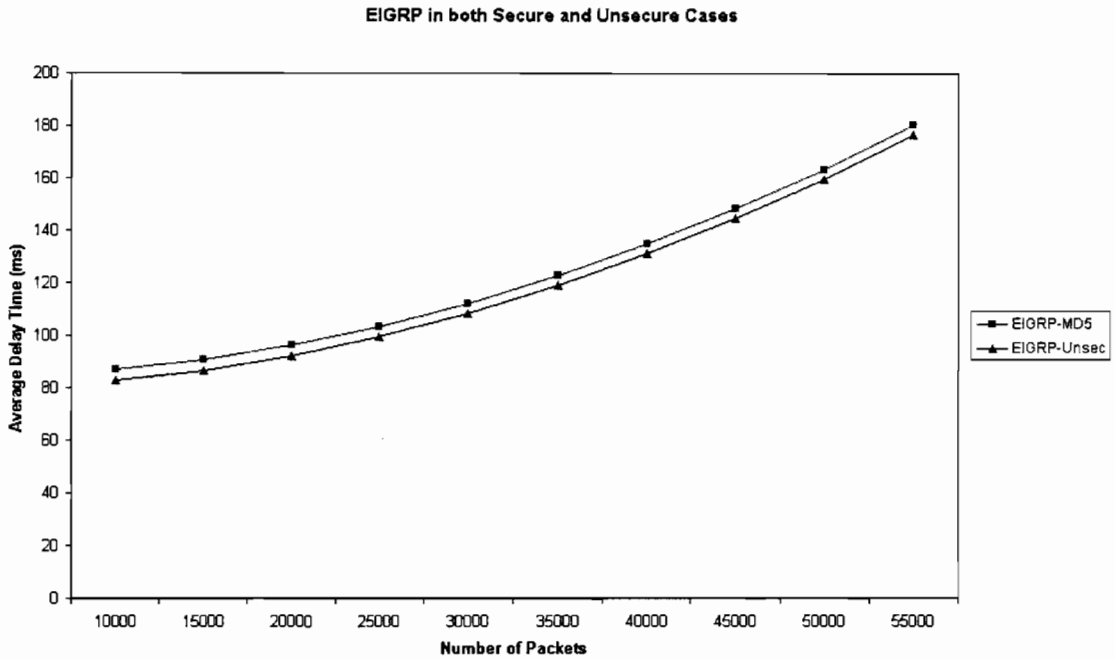


**Figure 5.1: Average Delay Time of Secured / No-secured EIGRP**

The results show the average delay time of EIGRP in secure mode is continuously larger than the EIGRP in unsecure mode due to many processes run that will delay the data passes to the server. Indeed, for both cases, the system shows an exponentially curve for lightly, moderately and extremely overloads.

### 5.2.2 Jitter of EIGRP in both Secured/Unsecured

Figure 5.2 shows the jitter with number of packets in the secured MD5 authentication case and the unsecured case for EIGRP routing protocol. The results show that during the system starts to moderately, MD5 is little bit highest than the unsecured case, when the system starts to extremely overload the jitter curve will is the same for the both secured and unsecured modes, because the system is going to steady state conditions.

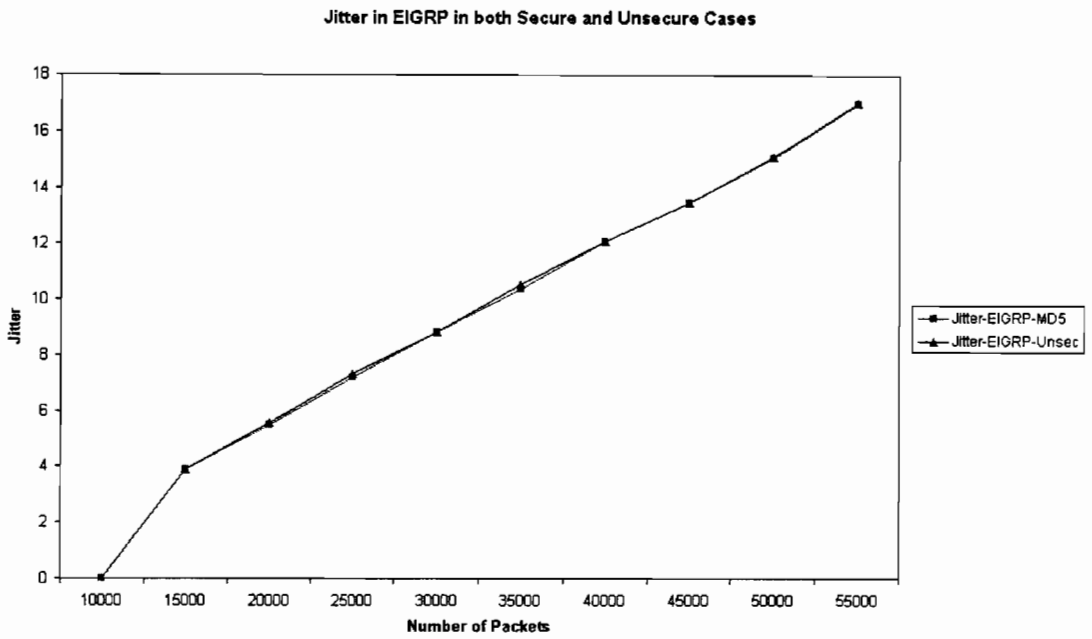


**Figure 5.2: Jitter of Secured / Unsecured EIGRP**

### 5.2.3 Overhead of EIGRP

Figure 5.3 shows the average delay overhead of the EIGRP routing protocol. The results show that when the system is lightly overloaded between 10,000-20,000 EIGRP gives the highest overhead values, when the system starts to moderately overloaded in 25,000-30,000 the EIGRP gives going to be a little overhead. Eventually, when the system is extremely overloaded the EIGRP gives the lowest overhead and the overhead will be almost in the steady state.

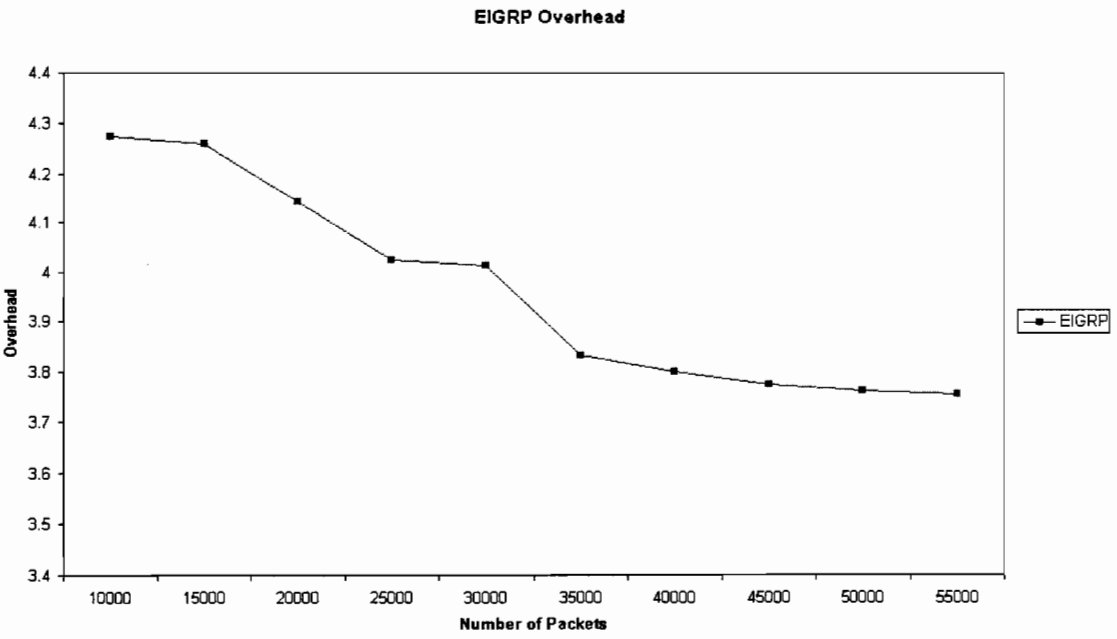


**Figure 5.3: EIGRP Overhead**

## 5.3 Routing Information Protocol version 2 (RIPv2)

In the following subsections, three graphs were plotted to evaluate the efficiency of the RIPv2 in both non-secured and secured MD5 authentication with respect to the average delay, jitter, and overhead respectively.

### 5.3.1 Average Delay Time of RIPv2 in both Secured & Unsecured

Figure 5.4 shows the average delay time with number of packets in the secured MD5 authentication case and unsecured case for RIPv2 routing protocols.

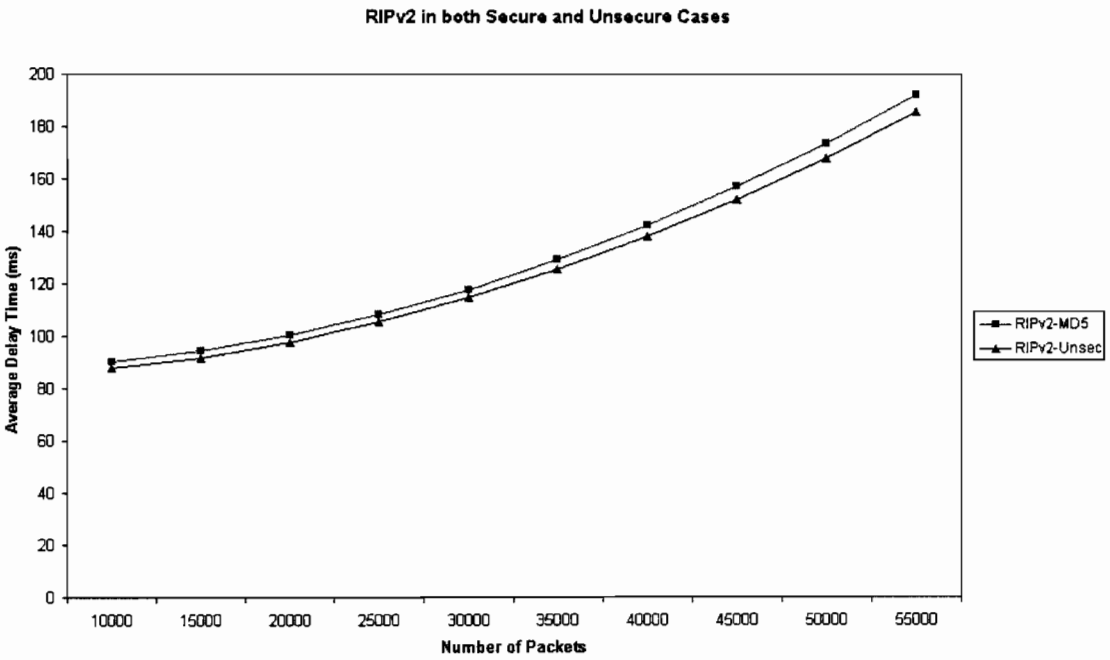


**Figure 5.4: Average Delay Time of Secured / Non-secured RIPv2**

The results show the average delay time of RIPv2 in secure mode is continuously larger than the RIPv2 in unsecure mode, because there are many processes run like producing a hash and check if hashes are the same or not, that will delay the data passes to the server. Indeed, for both cases, the system shows an exponentially curve for lightly, and moderately overloads, but in extremely overloaded the curse is going to highest value.

### 5.3.2 Jitter of RIPv2 in both Secured/Unsecured

Figure 5.5 shows the jitter with number of packets in the secured MD5 authentication case and the unsecured case for RIPv2 routing protocol. The results show that in the case of lightly loaded conditions, the jitter of RIPv2 in the secured and unsecured cases preserve the same jitter values. However, when the system starts to moderately overload with an approximate value of 4.2 and 20,000 packets, the RIPv2 in secured MD5 authentication case shows more exponentially curve when compared to the unsecured case, because there are many processes are running like producing a hash and check if the hashes are the same or not in addition to multicasting the entire routing table to all routers every 30 seconds.

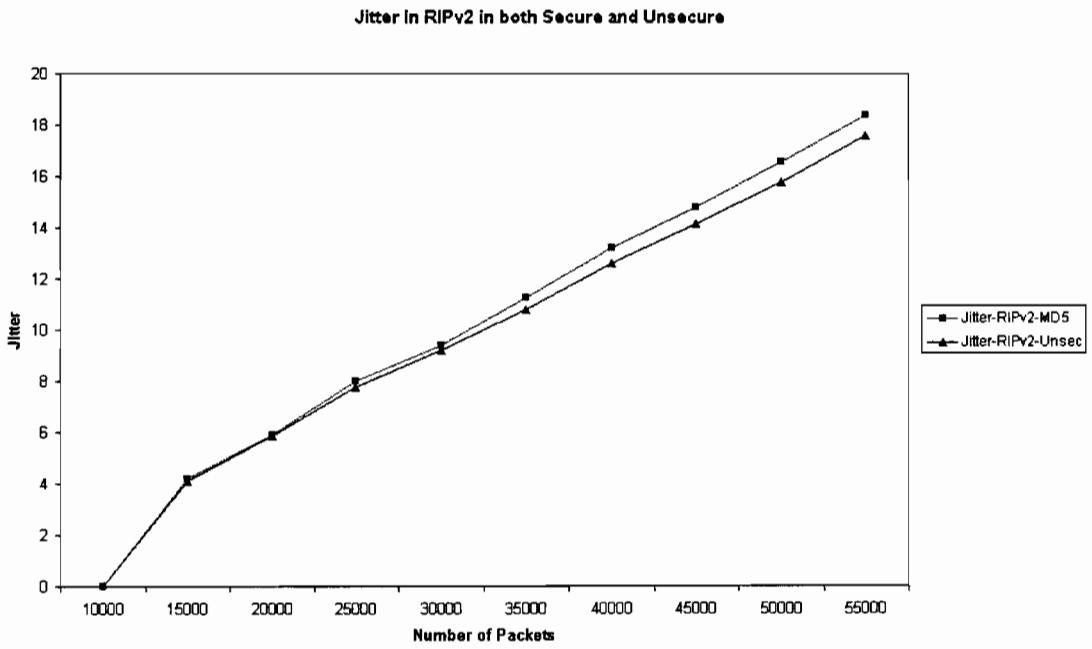


**Figure 5.5: Jitter of Secured / Non-secured RIPv2**

### 5.3.3 Overhead of RIPv2

Figure 5.6 shows the average delay overhead of the RIPv2 routing protocol. The results show that when the system is lightly overloaded, RIPv2 gives the lowest overhead values, when the system starts to moderately overloaded with 20,000 and 30,000 packets, the RIPv2 gives a bit more overhead. Eventually, when the system is extremely overloaded from 35,000 packets the RIPv2 starts to increase

exponentially, this is because of many processes are running in the router memory, in addition to multicasting the entire routing table to all routers every 30 seconds.

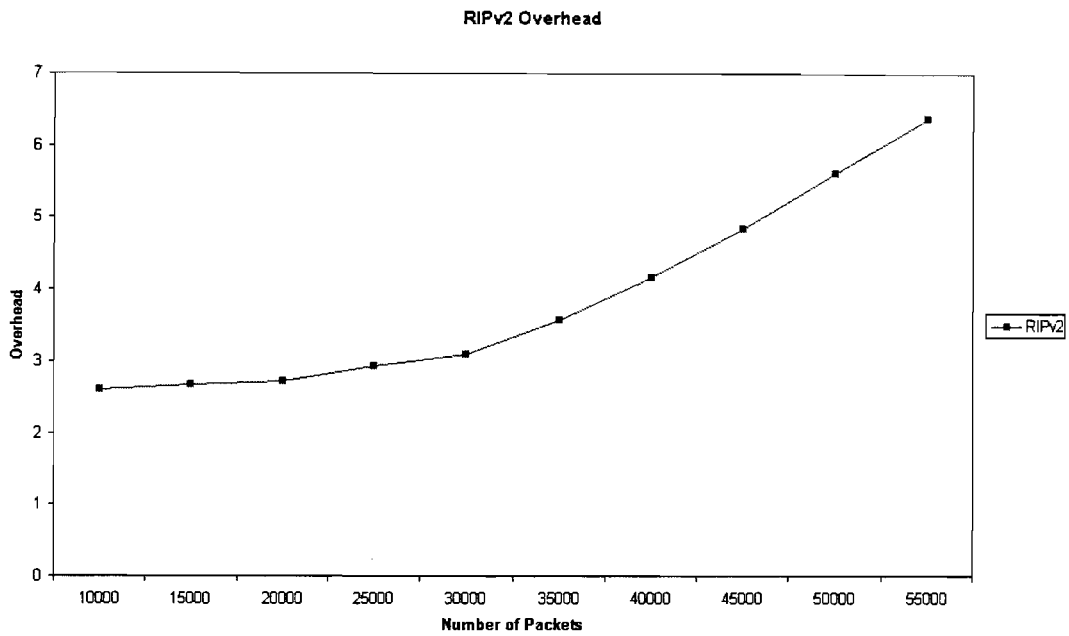


**Figure 5.6: Overhead of RIPv2**

## 5.4 Open Shortest Path First (OSPF)

In the following subsections, three graphs were plotted to evaluate the efficiency of the OSPF in both non-secured and secured MD5 authentication with respect to the average delay, jitter, and overhead respectively.

### 5.4.1 Average Delay Time of OSPF in both Secured/Non-Secured

Figure 5.7 shows the average delay time with number of packets in the secured MD5 authentication case and unsecured case for OSPF routing protocol. The results show that when the system is lightly overloaded the average delay time of OSPF in the unsecured is a bit smaller than the case of secured MD5 authentication in one. However, after then, when the system starts moderately overload the cure is shows exponentially increasing values little bit, when the system is extremely overload the OSPF MD5 case is showing more than the unsecured case, this is because of many processes are running in the router memory.

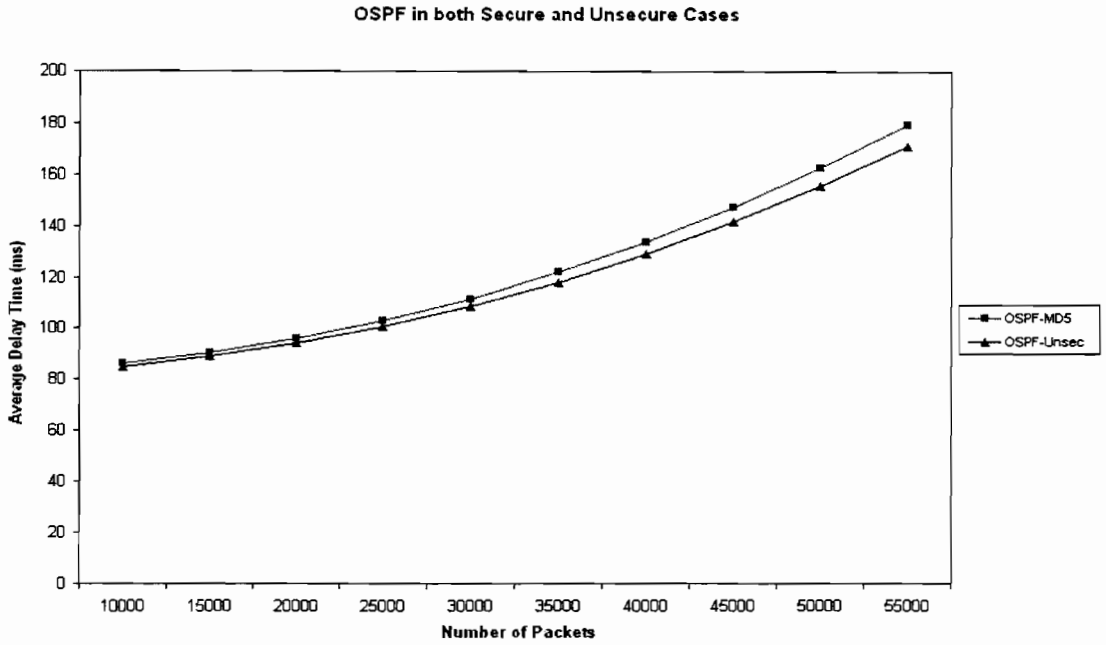


**Figure 5.7: Average Delay Time of Secured / Non-Secured OSPF**

### 5.4.2 Jitter of OSPF in both Secured/Unsecured

Figure 5.8 shows the jitter with number of packets in the secured MD5 authentication case and the unsecured case for OSPF routing protocol.

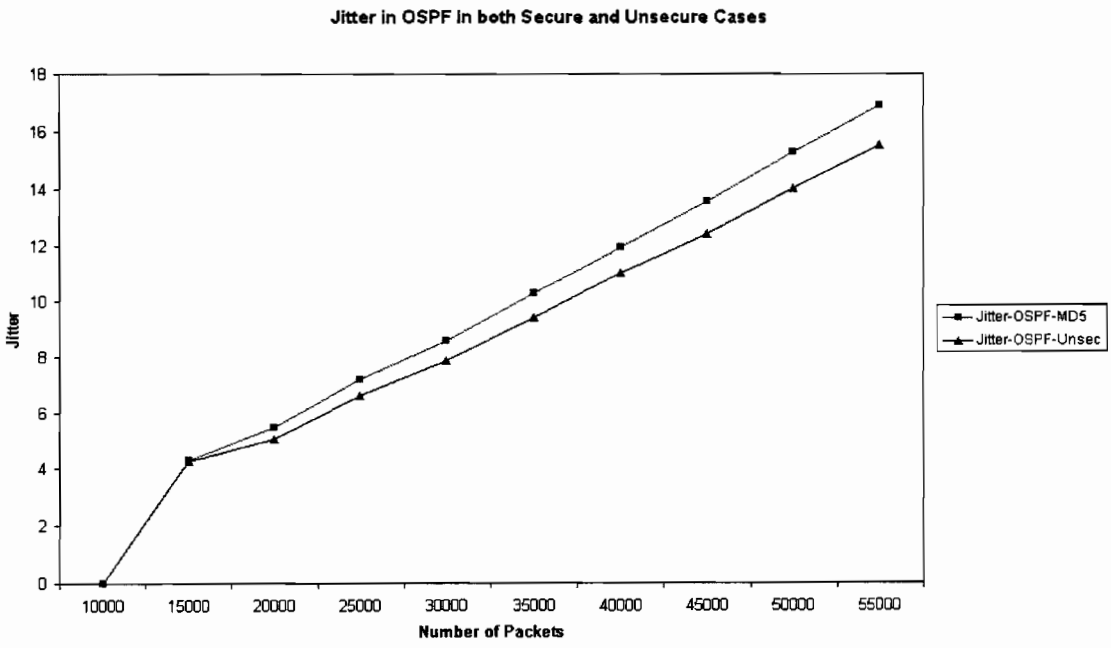


**Figure 5.8: Jitter of Secured / Non-Secured OSPF**

The results show that in the case of lightly loaded conditions, the jitter of OSPF in the secured MD5 authentication case is a little bit highest the unsecured case. However, when the system starts to moderately and extremely overloads the OSPF in secured MD5 authentication and unsecured cases, the cure starts an exponentially values with increasing jitter values in MD5 secured case, where the curve of that case in the average delay time was exponentially highest than the unsecured case, this is why, because of many processes are running in the router memory.

### 5.4.3 Overhead of OSPF

Figure 5.9 shows the average delay overhead of the OSPF routing protocol. The results show that when the system is lightly overloaded OSPF gives the lowest overhead, when the system starts to moderately overloaded the OSPF routing protocol gives almost the same overhead with an approximate value of 1.2 ms. Eventually, when the system is extremely overloaded, OSPF shows an exponentially overhead.

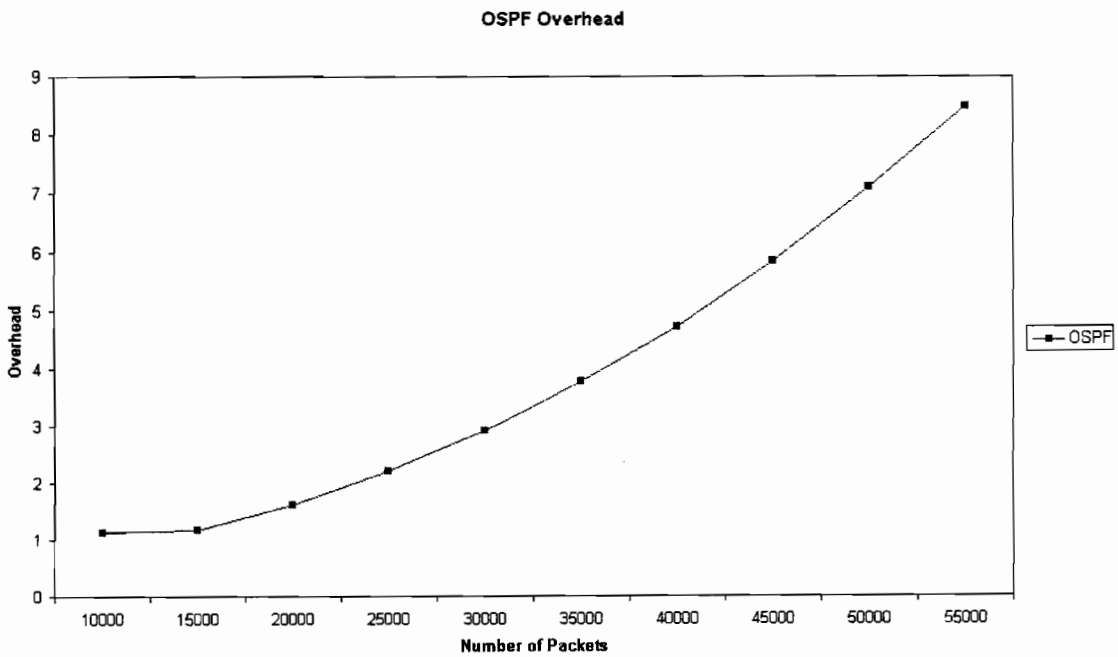


**Figure 5.9: Overhead of OSPF**

## 5.5 Total Analysis

Generally speaking, the thesis is combined from hardware and software perspectives. The proposed test-bed network model is deployed in the network research lab which means there is no other traffic on the network except the packet that is generated by the client and traverse through the model to the server, and the performance measures of interest can then be derived from the resulting probability vector. Due to space limitations, this section only describes key aspects of the proposed model. Particularly, the unsecured mode is coming from the default setup for each router in that model. On other hand, in the secured MD5 authentication mode the router will check the received packets for its valid authentication, then either it will pass or discard it.

In this subsection, there are nine graphs plotted to evaluate the average delay time, jitter and overhead with respect to the number of packets in both secured and unsecured routing modes. Various traffic loads described by total number of packets sent during the sessions of the ON periods have been plugged into the simulation model. Initially a total of 10,000 packets as a first traffic load incremented by 5,000 packets up to 55,000 packets have been used.

### 5.5.1 Unsecured Average Delay Time

Figure 5.10 shows the average delay time with number of packets in the unsecured case of EIGRP, RIPv2, and OSPF routing protocols. The results show the average delay time of RIPv2 is continuously larger than the other two routing protocols. However, when the system is lightly overloaded OSPF is larger than EIGRP until the 30,000 packets, when the system is moderately overloaded both OSPF and EIGRP gives the same results before the last one increase more when the system starts to extremely overloaded with 40000 packets processed, why, because EIGRP integrates the capabilities of link-state protocols (OSPF) into distance vector protocols.

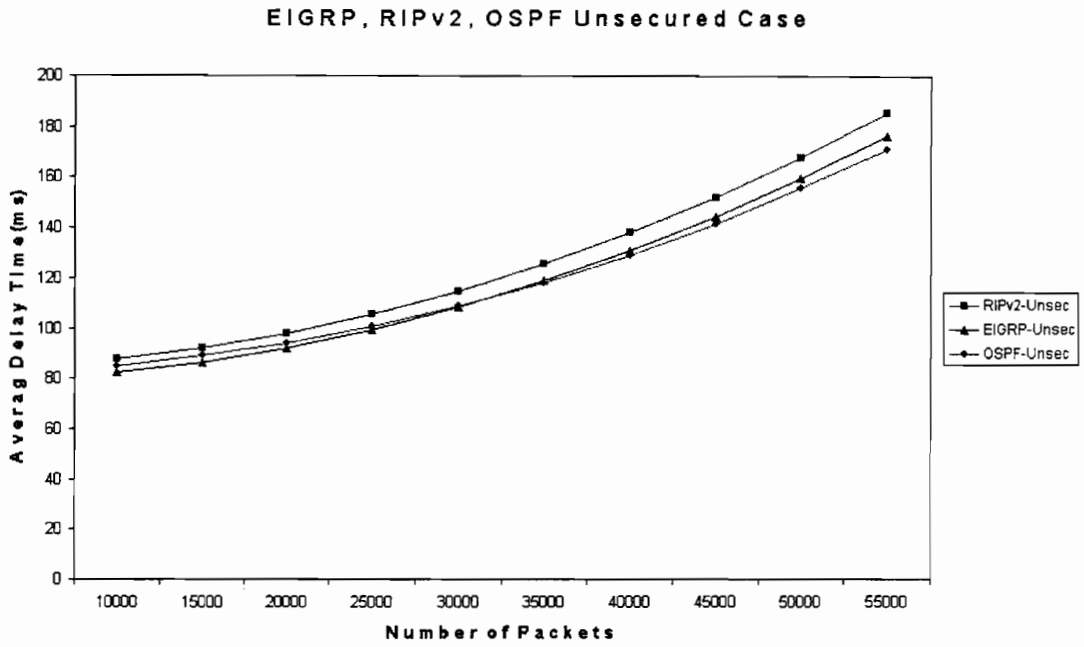


**Figure 5.10: Average Delay Time in Non-Secured Mode**

### 5.5.2 Secured MD5 Average Delay Time

Figure 5.11 shows the average delay time with number of packets in the secured MD5 authentication case of EIGRP, RIPv2, and OSPF routing protocols.

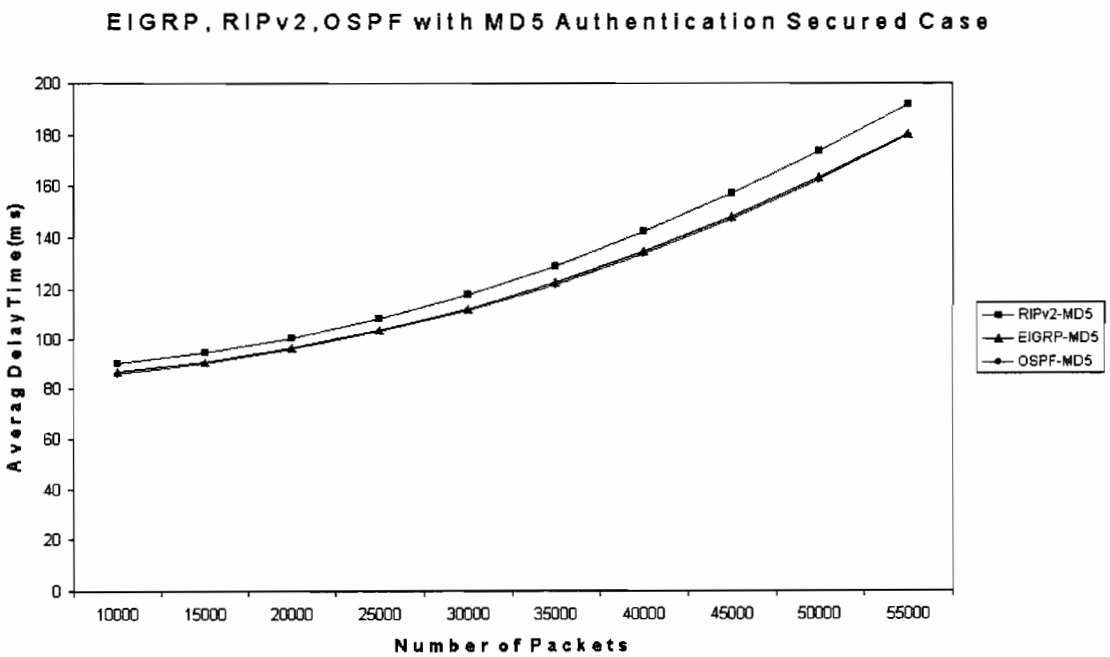


**Figure 5.11: Average Delay Time in Secured MD5 Authentication**

Figure 5.11 shows the average delay time with number of packets in the MD5 Authentication secured case of EIGRP, RIPv2, and OSPF routing protocols. The results show the average delay time of RIPv2 in the secured MD5 authentication case is continuously larger than the other two routing protocols, due to a many processes are running in the router memory, in addition to multicasting the entire routing table to all routers every 30 seconds. However, EIGRP gives a little bit highest result than OSPF with increase 1 ms. in EIGRP measurements. Why, this is due to the fact that OSPF is a link state which minimize the packets' processing delay time.

### 5.5.3 Unsecured of Jitter

Generally speaking, jitter is an important metric when considering the above routing protocols, and can be measured in a number of ways. The jitter measure used in this thesis is "cycle-to-cycle" jitter. This measure is taken by recording the difference in end-to-end delay of two successive packets of the same flow. For example, if a number packets arrive at the destination node having taken 90ms to traverse the network, and the following packet from the same flow takes 80ms, then the jitter for the second packet is 10ms. Figure 5.12 shows the jitter with number of packets in the unsecured case of EIGRP, RIPv2, and OSPF routing protocols.

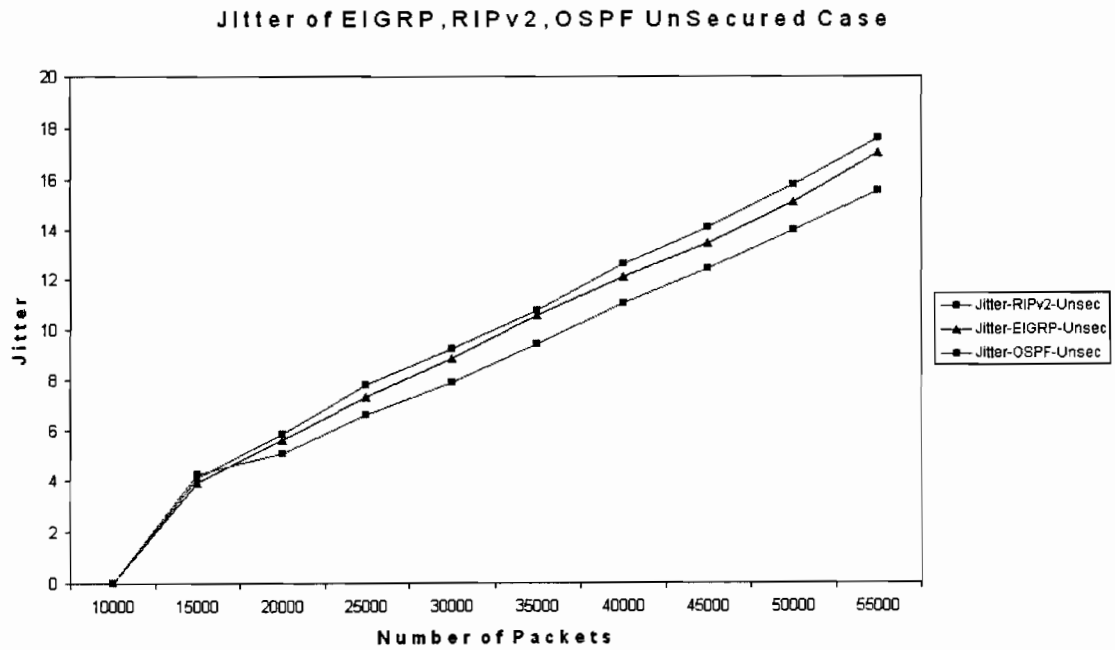


**Figure 5.12: Jitter in Non-Secured Mode**

Figure 5.12 shows the jitter with number of packets in unsecured case of EIGRP, RIPv2, and OSPF routing protocols. The results show that in the case of lightly loaded conditions, the three routing protocols preserve almost the same jitter value. However, when the system starts to moderately overloaded both RIPv2 and EIGRP show lager values when compared to the OSPF routing protocol. This is due to the fact that OSPF Why, this is due to the fact that OSPF is a link state which minimizes the packets' processing delay time and has the minimal average delay variation as shown earlier.

### 5.5.4 Secured MD5 Jitter.

Figure 5.13 shows the jitter with number of packets in the secured MD5 authentication case of EIGRP, RIPv2, and OSPF routing protocols. The results show that in the case of lightly loaded conditions with secured MD5 authentication case, the protocol who gives larger jitter vales is OSPF, then RIPv2 and lastly EIGRP. However, when the system starts to moderately overloaded the RIPv2 shows lager values when compared to the EIGRP and OSPF routing protocols which have a close values. This is due to the fact that both EIGRP (where integrates the capabilities of link-state protocols into distance vector protocols which minimizes the packets' processing delay time) and OSPF have the minimal average delay variation.

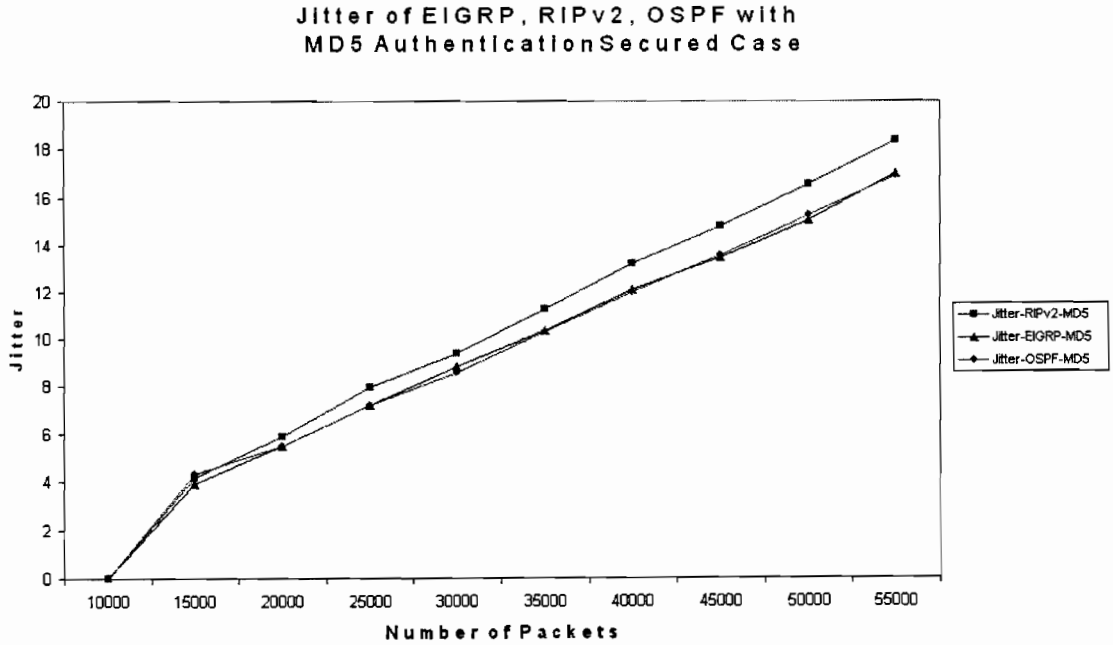


**Figure 5.13: Jitter in Secured MD5 Authentication Mode**

### 5.5.5 Overhead of EIGRP, RIPv2 and OSPF.

Figure 5.14 shows the average delay overhead of EIGRP, RIPv2, and OSPF routing protocols. The results show that when the system is lightly overloaded OSPF gives the lowest overhead while EIGRP gives the largest one. However, when the system starts to moderately overloaded at 35,000 packets the three routing protocols give almost the same overhead with an approximate value of 3.5 ms. Eventually, when the system is extremely overloaded, both RIPv2 and OSPF show an exponentially overhead while EIGRP remains almost stable.

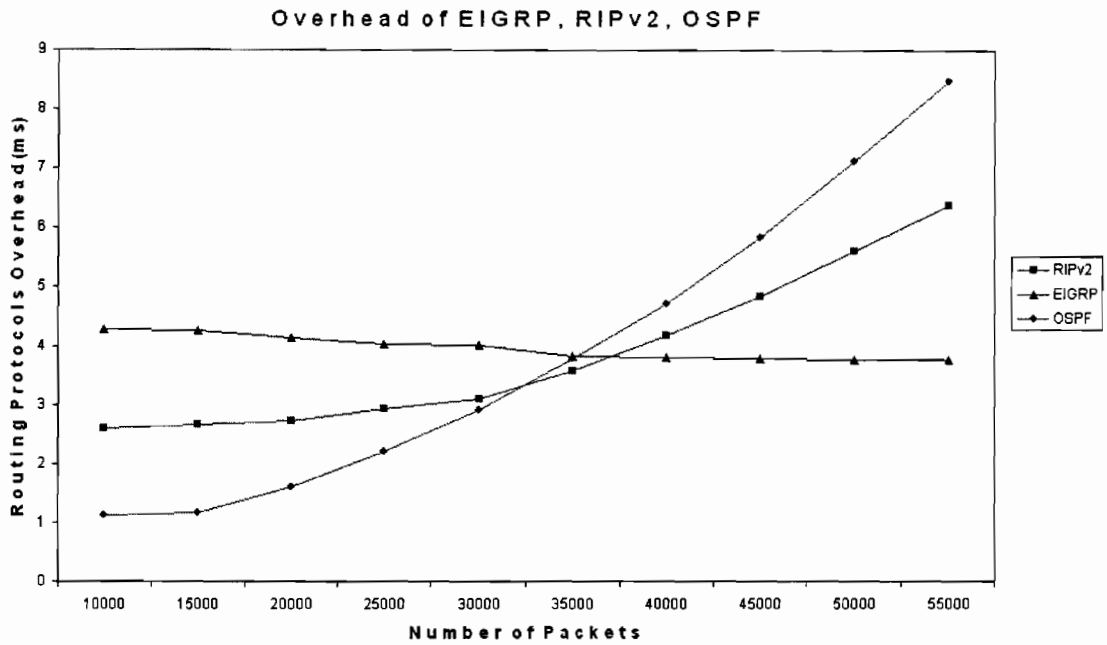


**Figure 5.14: Overhead of EIGRP, RIPv2, OSPF Routing Protocols**

## 5.6 Conclusions

In this chapter, we studied the impact of secured MD5 authentication versus un-secured for EIGRP, RIPv2, and OSPF routing protocols. The results obtained from the experiment showed that the average delay time and jitter in the secured case can become significantly larger when compared to the unsecured case even in steady state conditions. However, the EIGRP protocol shows the better performance by achieving the minimum overhead even when the system is extremely overloaded, why, because EIGRP integrates the capabilities of link-state protocols into distance vector protocols which minimizes the packets' processing delay time.

# Chapter 6

# Research Conclusions

## 6.1 Conclusion

In this thesis, a performance evaluation study for the Impact of MD5 authentication in the case of EIGRP, RIPv2, and OSPF routing protocols has been designed and implemented. The impacts of MD5 authentication are defined based on some sort of Internet-Draft published by Cisco Systems and applied on a proposed test-bed model combined to a Java-based Object-Oriented discrete-event simulator. The study is employed to calculate the average delay time, jitter and overhead of the above mentioned routing protocols. Thus, the main focus for this research is to evaluate the impact of MD5 authentication on these routing protocols mechanism.

The implementation of reliable Client/Server turned out to be the most challenging tasks, as these mechanisms called for modification to some very basic features of MD5 Impacts. Although, the implementation design of these mechanisms may seems to be quite straight forward, getting the in-depth insight into MD5 impacts, as well as the nature of these mechanisms, proved to be very difficult, especially with regards to the actual implementation.

Eventually, we presented three measurements in the test-bed network model to evaluate the performance of EIGRP, RIPv2 and OSPF routing protocols. The selected performance measures of interest were the average delay time, jitter, and overhead. By conducting the above three measurements and observing the collected results, it was extremely obvious that the EIGRP routing protocol performs overhead much better than other two RIPv2 and OSPF routing protocols. This can be clearly noticed through achieving the minimum average delay time, jitter, and overhead as our performance measures of interest. Moreover, from all the above we found that the impacts of MD5 Authentication on distance vector and link state routing protocols is almost the same values except RIPv2 due to its periodic updates for every 30 second.

## 6.2 Additional remarks

Doing research as well as the actual implementation with respect to this thesis has left a number of open questions which, due to time constraints, unfortunately must be left to future research and work. Some of these issues were discussed in Chapter 2.

A lot of time has been put into issues not directly related to the implementation. Such issues include learning some Java programming, learning how to configure Cisco routers, and other technical issues needed for this thesis.

Furthermore, a few papers on routing protocols, security and MD5, as well as general networking concepts, have been studied in order to gain an in-depth knowledge on the state-of-art technologies and research results. Some of these papers have influenced this thesis with regards to design choices and future work suggestions.

Being such a comprehensive and complex routing protocol, reading up on and understanding EIGRP, RIPv2 and OSPF proved to be very challenging. As RFC's and for the most part describes information for MD5 Authentication and routing protocols, both specifications had to be studied on order to acquire the in-depth knowledge needed for thesis.

## 6.3 Future work

During the process of evaluating the Impact of MD5 authentication on routing traffic, as described earlier in this thesis, new ideas and technical issues left an open research questions. However, due to our research scope constraints defined earlier for this master thesis these issues are suggested as future work.

In addition, the impact of IPv6 security on routing traffic for the cases of EIGRP, RIPv2 and OSPF routing protocols is our suggestion for future works on some issue related to the routing protocols security.

## References


[1] *Jeff Doyle, Jennifer DeHaven Carroll*, Routing TCP/IP, Volume II (CCIE Professional Development), Publisher: Cisco Press, Pub Date: April 11, 2001, ISBN: 1-57870-089-2.

[2] *Sackett, George,* Cisco Router Handbook. Blacklick, OH, USA, 1999.

[3] *Scott M. Ballew, 1997.* Managing IP Networks with Cisco Routers. 1st Edn., O'Reilly Media, Inc., Lawrence, MA, United States, pp: 352. ISBN: 1565923200.

[4] *Ravi Malhotra*, IP Routing, O'Reilly Online Catalog, O'Reilly & Associates Inc. *January 2002* accessed 4/3/2009

[5] *Merike Kaeo,* Designing Network Security Second Edition, Publisher: Cisco Press, Pub Date: October 30, 2003, ISBN: 1-58705-117-6

[6] *V. Anand and K. Chakrabarty,* Cisco IP Routing Protocols: Troubleshooting Techniques, Charles River Media 2004 - ISBN: 1584503416.

[7] *Gert De Laet, Gert Schauwers*, Network Security Fundamentals, Publisher Cisco Press, September 08, 2004 ISBN: 1-58705-167-2.

[8] *Wendell Odom,* CCNA INTRO Exam Certification Guide, Copyright© 2004 Cisco Systems, Inc., Published by: Cisco Press ISBN: 1-58720-094-5

[9] *Wendell Odom,* CCNA ICND Exam Certification Guide, Copyright© 2004 Cisco Systems, Inc. Published by: Cisco Press ISBN: 1-58720-083-x

[10] *Sam Halabi and Danny McPherson,* Internet Routing Architectures, Second Edition August 23, 2000 Publisher: Cisco Press, ISBN: 1-57870-233-X.

[11] *Jeff Doyle, Jennifer Carroll, Publisher,* CCIE Professional Development Routing TCP/IP, Volume I, Second Edition, Cisco Press, Pub Date: October 19, 2005; ISBN: 1-58705-202-4

[12] *Ivan Pepelnjak*, EIGRP-Network-Design-Solutions, Publisher: Cisco Press, Pub. Date: Jan 15, 2000.

[13] *Fung, K. T. (Kwok T.),* Network Security Technologies, Second Edition, Date: 2004, ISBN 0-8493-3027-0.

[14] *Merike Kaeo,* Designing Network Security Second Edition, Publisher: Cisco Press, Pub Date: October 30, 2003, ISBN: 1-58705-117-6

[15] http://www.cisco.com/en/US/docs/ios/12_0/np1/configuration/guide/1crip.html accessed 22/4/2009.

[16] http://www.oreilly.com/catalog/iprouting/chapter/ch04.html accessed 22/4/2009

[17] http://www2.rad.com/networks/1995/ospf/ospf.htm accessed 22/4/2009

[18] *B. Albrightson and J.J. Garcia-Luna-Aceves and J. Boyle*, EIGRP – A fast routing protocol based on distance-vectors, in: *Proceeding of Networld/Interop '94*, Las Vegas, NV, May 1994. accessed 11/1/2009

[19] *Talal M. Jaafar, George F. Riley, Dheeraj Reddy and Dana Blair*, SIMULATION-BASED ROUTING PROTOCOL PERFORMANCE ANALYSIS – A CASE STUDY, in the Proceedings of the 2006 Winter Simulation Conference 2006 IEEE 1-4244-0501-7/06, USA. Accessed 11/11/08

[20] *Ivan Pepelnjak 2000*, EIGRP network design solutions, Cisco Press ISBN 1578701651, Release Date 15 January 2000.

[21] *Mohamed G. Gouda, and Marco Schneider,* Maximizable Routing Metrics, in the proceeding of IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 11, NO. 4, AUGUST 2003 663. accessed 10/4/2009.

[22] *Nigel Houlden, Vic Grout, John McGinn and John Davies*, Extended End-to-End Cost Metrics for Improved, Dynamic Route Calculation, Proceedings of the 6th International Network Conference (INC 2006), University of Plymouth, 11-14 July 2006, pp89-96. accessed 10/3/2009.

[23] *Franck Le, Geoffrey G. Xie, Dan Pei, JiaWang and Hui Zhang,* Shedding Light on the Glue Logic of the Internet Routing Architecture, *in proceeding of SIGCOMM'08, August 17–22, 2008, Seattle, Washington, USA.* accessed 10/3/2009

[24] *Gary Scott Malkin,* RIP Version 2 Carrying Additional Information, Xylogics, Inc., November 1994 - RFC1723.

[25] *Tao Wan Evangelos Kranakis P.C. van Oorschot,* "S-RIP: A Secure Distance Vector Routing Protocol," *in the Proceedings of Applied Cryptography and Network Security (academic track)*, Yellow Mountain, China. June 8-11 2004. accessed 10/3/2009

[26] *Dan Pei, Dan Massey and Lixia Zhang*, "Detection of Invalid Routing Announcements in RIP Protocol," published *in GLOBECOM IEEE*, volume 3, Dec 2003. accessed 10/3/2009

[27] *Dan Pei and Lixia Zhang, Dan Massey*, "A Framework for Resilient Internet Routing Protocols," *IEEE Network*, March/April 2004. accessed 10/2/2009

[28] *Abdelaziz Babakhouya, Yacine Challal, Abdelmadjid Bouabdallah, SaYd Gharout*, "S-DV: A new approach to Secure Distance Vector routing protocols",. 10.1109/SECCOMW, Aug. 28 -Sept. 1, 2006, Baltimore, MD, ISBN: 1-4244-0423-1, IEEE 2006. accessed 10/3/2009

[29] *David A. Maltz, Geoffrey Xie, Jibin Zhan, Hui Zhang, G´ısli Hj ´almt´ysson, Albert Greenberg,* Routing Design in Operational Networks: A Look from the Inside, in proceeding of SIGCOMM'04, August 30–Sept. 3, 2004, Portland, Oregon, USA. accessed 15/3/2009

[30] *David Bauer, Murat Yuksel, Christopher Carothers and Shivkumar Kalyanaraman,* A Case Study in Understanding OSPF and BGP Interactions Using Efficient Experiment Design, in the Proceedings of the 20th Workshop on Principles of Advanced and Distributed Simulation (PADS'06) 2006 IEEE. accessed 10/1/2009

[31] *Bae, Sang; Henderson, Thomas R.,* Traffic Engineering with OSPF Multi-Topology Routing, in the proceedings of Military Communications Conference, 2007, MILCOM, IEEE 29-31 Oct. 2007 Page(s):1 - 7. accessed 10/2/2009

[32] *George F. Riley, Dheeraj Reddy*, Simulating Realistic Packet Routing Without Routing Protocols, in the Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS'05) 1087-4097/05, IEEE 2005. accessed 10/3/2009

[33] *Ching-Chuan Chiang, Chinyi. Chen, Dah-Lih Jeng, Shuenn-Jyi Wang, and Ying-Kwei Ho*, The Performance and Security Evaluations of Internet Routing Protocols, published in Journal of Informatics & Electronics, Vol.2, No.2, pp.21-27, March 2008. accessed 10/3/2009

[34] *F. Baker and R. Atkinson*, January 1997. RIP-2: MD5 Authentication. IETF RFC2082, Cisco Systems, United States, no of Pages 12. http://portal.acm.org/citation.cfm?id=RFC2082 accessed 10/3/2008

[35] *John Moy,* OSPF Version 2, Ascend Communications, Inc. Network Working Group, April 1998. RFC 2328. accessed 10/5/2008

[36] *Fred Baker,* DRAFT OSPF MD5 Authentication, Advanced Computer Communications, September October 1994, draft-ietf-ospf-md5-01.txt. accessed 10/4/2008

[37] *Emanuele Jones and Olivier Le Moigne,* OSPF Security Vulnerabilities Analysis, Alcatel Company Canada, Network Working Group, June 16, 2006. (draft-jones-ospf-vuln-02.txt). accessed 10/1/2008

[38] *Ronald L. Rivest*, Massachusetts Institute of Technology, MD5 Message-Digest Algorithm, *RFC 3121*, MIT Laboratory for Computer Science, April 1992 accessed 20/1/2008

[39] *R. Rivest.* The MD5 Message-Digest Algorithm, *IETF RFC 1321, MIT Laboratory for Computer Science and RSA Data Security Inc., April 1992. http://portal.acm.org/citation.cfm?id=RFC1321* accessed 15/3/2008

[40] *Khalid Abu Al-Saud, Hatim Mohd Tahir, Adel Elzoghabi, Mohammad Saleh,* Performance Evaluation of Secured versus Non-Secured EIGRP Routing Protocol, Proceeding of WORLDCOMP '08, Las Vegas, NV, July 14-17, 2008.

[41] *Khalid Abu Al-Saud, Hatim Mohd Tahir, Moutaz Saleh and Mohammad Saleh,* Impact of MD5 Authentication on Routing Traffic for the Case of: EIGRP, RIPv2 & OSPF, Proceeding in Journal of Computer Science, 244, 5th Avenue, Number S-207, New York, NY 10001, November 2008.

[42] *Ramaswamy Chandramouli, Tim Grance, Rick Kuhn, Susan Landau,* Toward Secure Routing Infrastructures, Proceedings of the IEEE SECURITY & PRIVACY 2006, pages 84-78, accessed 25/1/2008.

[43] *Bradly R. Smith and J.J. Garcia-Luna*, "Securing the Border Gateway Routing Protocol," Proceedings of the ISOC Symposium on Network and Distributed System Security '97, February 11, 1997. accessed 25/2/2008.

[44] *C.-T. Huang E. N. Elnozahy M. G. Gouda,* Hop Integrity and the Security of Routing Protocols, 2002, accessed 10/12/2007.

[45] *Deepakumara, J. H.M. Heys and R. Venkatesan.* FPGA implementation of MD5 hash algorithm. In the Proceedings of IEEE Canadian Conference on Electrical and Computer Engineering CCECE, May 13-16, 2001, Toronto, Ontario, Canada, pp: 919-924. accessed 15/3/2008

[46] *Dijiang Huang, Amit Sinha and Deep Medhi, 2003.* A double authentication scheme to detect impersonation attack in link state routing protocols. In the Proceedings of IEEE International Conference on Communications (ICC), pages 1723-1727, Vol. 3, May 11-15, Anchorage, Alaska 2003. accessed 10/3/2007.

[47] *F. Baker and R. Atkinson, October 1994.* OSPF MD5 Authentication, draft-ietf-ospf-md5-02.txt, Naval Research Laboratory, no Pages 11. accessed 18/2/2008.

[48] http://en.wikipedia.org/wiki/MD5 last visit 22/4/09

[49] *C. Demichelis and P. Chimento*, RFC: 3393, November 2002, IP Packet Delay Variation Metric for IP Performance Metrics (IPPM). accessed 15/3/2008

[50] *Schmitd, J.W., and Taylor, R.E. 1970. Simulation and Analysis of Industrial Systems*. Homewood, Illinois.

[51] *Averill, M.L. and Kelton, W.D. 2000. Simulation Modeling and Analysis*. Mc Graw Hill.

[52] *Jain, R. 2000.* The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurements, Simulation and Modeling. John Wiley, NewYork.

[53] http://www.cisco.com/en/US/products/hw/routers/ps221/index.html acc. 22/4/09

[54] Cisco-1721-1720datasheet.pdf, accessed 12/12/2007.

[55] http://www.cisco.com/univercd/cc/td/doc/product/lan/c3550/1216ea1/3550hig/35cable.htm last visit 22/4/2009

[56] *Kwok T. Fung*, Network Security Technologies, CRC Press, 2nd Edition, 2005.

[57] *Merike Kseo,* Designing Network Security, second edition, Cisco Press, October 30, 2003.

[58] *Kevin J. Healy, Richard A. Kilgore*, A JAVA-BASED PROCESS SIMULATION LANGUAGE, Proceeding of the 1997 Winter Simulation Conference, ed. S. Andradottir, K. J. Healy, D. H. Withers, and B. L. Nelson, accessed 15/3/2008

[59] *R. Albrightson and J.J. Garcia-Luna-Aceves and J. Boyle*, EIGRP – A fast routing protocol based on distance-vectors, in: *Proceeding of Networld/Interop '94*, Las Vegas, NV, May 1994. accessed 10/6/2007

[60] *Reinhard Finstenvalder*, A generic client/server architecture for distributed Web-based simulation experimentation, Computer-Aided Control System Design, 2000 (CACSD2000). IEEE International Symposium on Volume, Issue, 2000 Pages: 185 – 189 Digital Object Identifier accessed 10/4/2007.

[61] *N. Honn, N. Hohn, D. Veitch, K. Papagiannaki and C. Diot, 2004.* Bridging router performance and queuing theory. In Proceeding of the ACM SIGMETRICS/Performance'04, June 12–16, 2004, New York, NY, USA, pp:

355-366. http://portal.acm.org/citation.cfm?id=1005686.1005728. accessed 10/3/2008

[62] *Imad Antonios, Lester Lipsky 2002.* A performance model of user delay in on/off heavy-tailed traffic. In the Proceedings of the 2002 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS 2002), San Diego, CA, July, 2002. accessed 15/4/2008

[63] www.karjasoft.com/files/clocksync/ClockSync1.0.0.exe , last visit 22/4/09.