Volume 4 ▪ Issue 5 ▪ December 2012

Editor-in-Chief
Dr. Min Song

INTERNATIONAL JOURNAL OF

COMPUTER NETWORKS (IJCN)

ISSN : 1985-4129

Publication Frequency: 6 Issues / Year

CSC PUBLISHERS
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF COMPUTER NETWORKS (IJCN)

VOLUME 4, ISSUE 5, 2012

EDITED BY
DR. NABEEL TAHIR

ISSN (Online): 1985-4129

International Journal of Computer Networks (IJCN) is published both in traditional paper form and in Internet. This journal is published at the website http://www.cscjournals.org, maintained by Computer Science Journals (CSC Journals), Malaysia.

IJCN Journal is a part of CSC Publishers
Computer Science Journals
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF COMPUTER NETWORKS (IJCN)

Book: Volume 4, Issue 5, December 2012
Publishing Date: 31-12-2012
ISSN (Online): 1985-4129

IJCN Journal is a part of CSC Publishers
http://www.cscjournals.org

Published in Malaysia

Typesetting: Camera-ready by author, data conversation by CSC Publishing Services – CSC Journals, Malaysia

CSC Publishers, 2012

EDITORIAL PREFACE

The International Journal of Computer Networks (IJCN) is an effective medium to interchange high quality theoretical and applied research in the field of computer networks from theoretical research to application development. This is the *fifth* Issue of Volume *four* of IJCN. The Journal is published bi-monthly, with papers being peer reviewed to high international standards. IJCN emphasizes on efficient and effective image technologies, and provides a central for a deeper understanding in the discipline by encouraging the quantitative comparison and performance evaluation of the emerging components of computer networks. Some of the important topics are ad-hoc wireless networks, congestion and flow control, cooperative networks, delay tolerant networks, mobile satellite networks, multicast and broadcast networks, multimedia networks, network architectures and protocols etc.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Starting with Volume 5, 2013, IJCN aims to appear with more focused issues. Besides normal publications, IJCN intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

IJCN give an opportunity to scientists, researchers, engineers and vendors to share the ideas, identify problems, investigate relevant issues, share common interests, explore new approaches, and initiate possible collaborative research and system development. This journal is helpful for the researchers and R&D engineers, scientists all those persons who are involve in computer networks in any shape.

Highly professional scholars give their efforts, valuable time, expertise and motivation to IJCN as Editorial board members. All submissions are evaluated by the International Editorial Board. The International Editorial Board ensures that significant developments in computer networks from around the world are reflected in the IJCN publications.

IJCN editors understand that how much it is important for authors and researchers to have their work published with a minimum delay after submission of their papers. They also strongly believe that the direct communication between the editors and authors are important for the welfare, quality and wellbeing of the journal and its readers. Therefore, all activities from paper submission to paper publication are controlled through electronic systems that include electronic submission, editorial panel and review system that ensures rapid decision with least delays in the publication processes.

To build its international reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJCN. We would like to remind you that the success of our journal depends directly on the number of quality articles submitted for review. Accordingly, we would like to request your participation by submitting quality manuscripts for review and encouraging your colleagues to submit quality manuscripts for review. One of the great benefits we can provide to our prospective authors is the mentoring nature of our review process. IJCN provides authors with high quality, helpful reviews that are shaped to assist authors in improving their manuscripts.

Editorial Board Members
International Journal of Computer Networks (IJCN)

EDITORIAL BOARD

TABLE OF CONTENTS

Volume 4, Issue 5, December 2012

Pages

Application of N jobs M machine Job Sequencing Technique for MPLS Traffic Engineering

Punit Kumar Singh *punit20.singh@gmail.com*
Department of Computer Science and Engineering
M.M.M Engineering College, Gorakhpur-273010
India

Dr. Rakesh Kumar *rkiitr@gmail.com*
Department of Computer Science and Engineering
M.M.M Engineering College, Gorakhpur-273010
India

Abstract

This paper discusses Traffic Engineering with Multi-Protocol Label Switching (MPLS) in an Internet Service Provider's (ISP) network. In this paper, we first briefly describe MPLS, Constraint-based Routing, MPLS-TE, N jobs M machine Job sequencing technique and how to implement the job sequencing technique for Multi-Protocol Label Switching Traffic Engineering. And also improve the quality of service of the network, using this technique firstly reduce the congestion for traffic engineering; minimize the packet loss in complex MPLS domain. In small network packet loss is negligible. We used NS2 discrete event simulator for simulate the above work.

Keywords: Traffic Engineering, Multi-Protocol Label Switching, Constraint based routing, N jobs M machine Job Sequencing Technique, Qos, MPLS-TE.

1. INTRODUCTION

Traffic Engineering is the process of controlling how traffic flows through one's network so as to optimize resource utilization and network performance [1, 2, 3]. Traffic Engineering is needed in the Internet mainly because current IGPs always use the shortest paths to forward traffic. Using shortest paths conserves network resources, but it may also cause the following problems.

i) The shortest paths from different sources overlap at some links, causing congestion on those links.
ii) The traffic from a source to a destination exceeds the capacity of the shortest path, while a longer path between these two routers is under-utilized.

There is a debate of whether network capacity will one day become so cheap and abundant that these two problems will be eliminated. This debate is beyond the scope of this paper. Here we simply note that currently all ISPs have the above problems. By performing Traffic Engineering in their networks, ISPs can greatly optimize resource utilization and network performance. Revenue can be increased without large investment in upgrading network infrastructure. In order to do Traffic Engineering effectively, the *Internet Engineering Task Force* (IETF) introduces MPLS [4], Constraint-based Routing [6] and N jobs M machine Sequencing Technique [7]. They are briefly reviewed in this section.

1.1 MPLS

We are now going to discuss the basics of MPLS protocol. This protocol was designed by Internet Engineering Task Force (IETF) and its specifications were given in RFC 3031(Request for Comments). Now we are going to give the basics of this protocol next, in form of points.

a) When MPLS protocol is implemented in any network or subnet then two special routers namely ingress and egress routers are imparted in it. See figure 1 below.

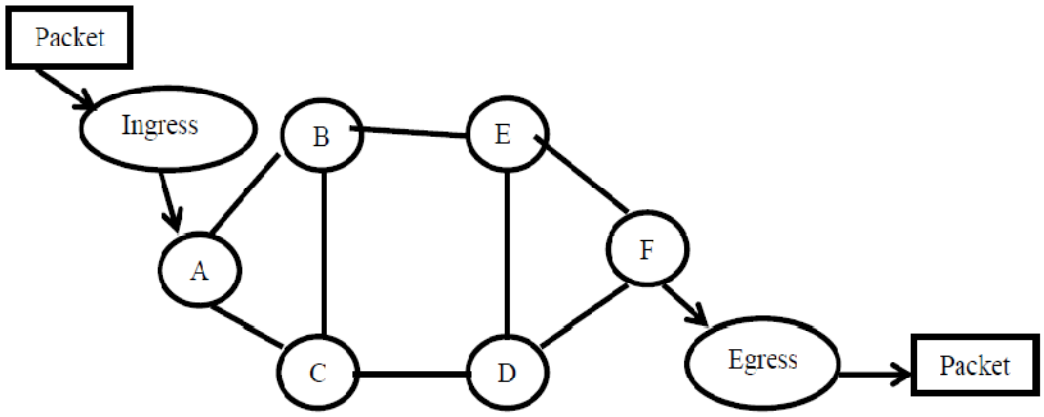

Figure 1.1: showing working of MPLS subnet with Egress and Ingress router attached

b) Then whenever packet has to move inside the MPLS subnet then it enters from the Ingress router and whenever any packet has to move out from the MPLS subnet then it moves out from the egress router.

c) Now whenever any packet has to move from source to destination through MPLS subnet then it is subjected to the ingress router first.

d) Then ingress router adds a tag or label which is known as MPLS label to it and switches it inside the MPLS subnet (see fig1).

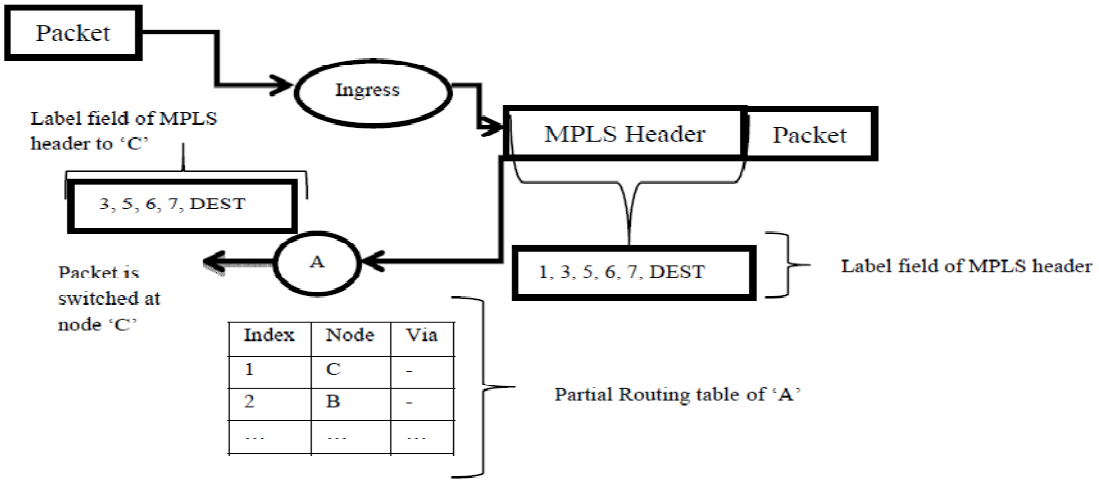

Figure 2: Showing the add MPLS header by ingress router and how to move from A to C.

e) Then using this MPLS label (Complete label format is shown in figure 3) this packed moves from one node to another within the MPLS network. Whenever this packet moves from one node to another then the node updates the MPLS header accordingly so it could easily be switched from one node to another until it don't reaches the destination node or egress router. For example see figure 2 in this we have shown Label field of MPLS header according to it packet has to move from ingress router towards destination. Now when packet reaches to ingress router it adds an MPLS tag in it and switches it to node say A. Now this node A has a routing table in it whose partial version is shown in figure 2. Now from label field first digit is read which is index 1. This index is searched in the routing table of node A and packet is switched to node C which is correspondent node at index 1. Moreover MPLS header is also modified by node A as shown in figure 2.

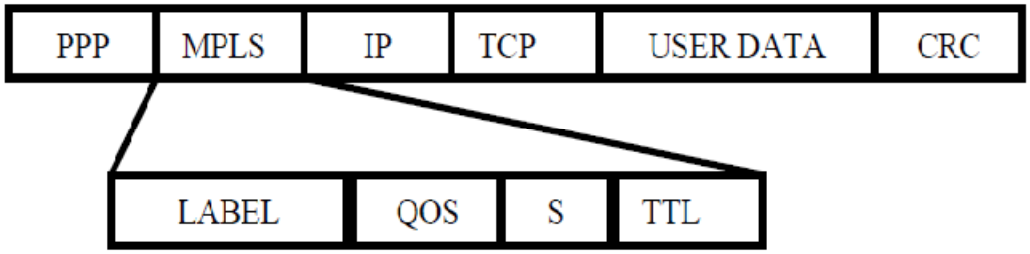

Figure 3: Showing the MPLS header Format

f) If the destination node is present inside the MPLS network then ok. Packet is switched there. But if destination node lies outside the MPLS network then packet reaches the egress router and there egress router removes the MPLS tag from it and routes the packet out of the MPLS network towards the destination.

1.2 Constraint Based Routing:

Constraint-based Routing (CBR) computes routes that are subject to constraints such as bandwidth and administrative policy. Because Constraint-based Routing considers more than network topology in computing routes, it may find a longer but lightly loaded path better than the heavily loaded shortest path. Network traffic is hence distributed more evenly. For example in Fig. 2, the shortest path between router A and router C is through link A-C with IGP metric *m*=1. But because the resolvable bandwidth on the shortest path is only (622-600) = 22 Mbps, when Constraint based Routing tries to find a path for an LSP of 40 Mbps, it will select path A-B-C instead, because the shortest path does not meet the bandwidth constraint.

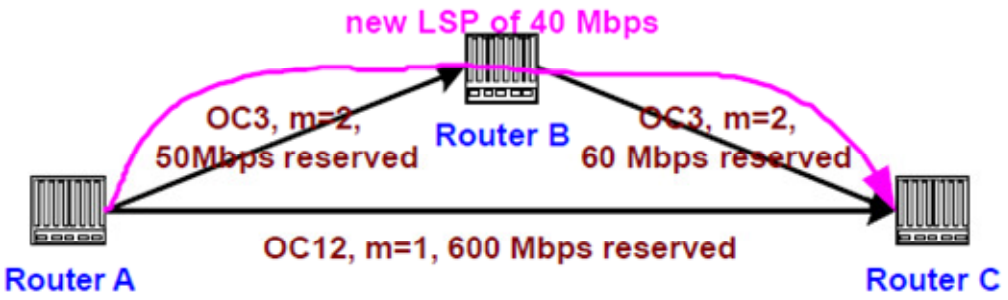

Figure 4: Constraint Based Routing

It should be noted that the reservable bandwidth of a link is equal to the maximum reservable bandwidth set by network administrators minus the total bandwidth reserved by LSPs traversing the link. It does not dependon the actual amount of available bandwidth on that link. For example, if the maximum reservable bandwidth of a link is 155 Mbps, and the total bandwidth reserved by LSPs is 100 Mbps, then the reservable bandwidth of the link is 55 Mbps, regardless of whether the link is actually carrying 100 Mbps of traffic or more or less. In other words, Constraint-based Routing does not compute LSP paths based on instantaneous residual bandwidth of links. This reduces the probability of routing instability [6]. Constraint-based Routing can be online or offline. With online Constraint-based Routing, routers may compute paths for LSPs at any time. With offline Constraint-based Routing, an offline server computes paths for LSPs periodically (hourly/daily). LSPs are then configured to take the computed paths.

1.3 Generic Issues of Designing an MPLS System for Traffic Engineering

To build an MPLS system for Traffic Engineering, the following design parameters must be determined:

1. The geographical scope of the MPLS system;
2. The participating routers;
3. The hierarchy of MPLS system;
4. The bandwidth requirement of the lsps;
5. The path attribute of the lsps;
6. The priority of the lsps;
7. The number of parallel lsps between each endpoint pair;
8. The affinity of the lsps and the links;
9. The adaptability and resilience attributes of the lsps.

The process of deciding the scope of an MPLS system is driven by administrative policy. Specifically, if the network architecture of a region is irregular (as opposed to the regular architecture showed in Fig. 4), or the capacity of a region is tight, then the region should be included in the MPLS system.

The second step is to decide the participating routers in the MPLS system, i.e., the ingress LSRs, the transit LSRs and the egress LSRs. This should also be guided by the administrative policy. Network administrators may want to forbid some routers from participating in the MPLS system for some reason, for example, because those routers cannot be trusted or because those routers do not have enough processing power and/or memory capacity. Another factor for consideration is the tradeoff between the number of LSPs and efficiency of the links. More ingress and egress LSRs mean more LSPs and thus higher LSP-routing complexity. But because the average size (bandwidth requirement) of the LSPs is smaller, Constraint-based Routing has more flexibility in routing the LSPs. Higher link efficiency may be achieved. After the LSRs are decided, network administrators need to decide the hierarchy of the MPLS system. One alternative is to fully mesh all LSRs, resulting in a single layer of LSPs. For large ISPs, there can be hundreds of LSRs. A full mesh will result in a huge MPLS system. Another alternative is to divide one's network into multiple regions. LSRs in each region are meshed. This forms the first layer of LSPs. Some selected LSRs from each region, for example the core routers, are also fully meshed to form the second layer of the LSPs. This hierarchical design can significantly reduce the number of LSPs in the network, and hence the associated processing and managing overhead. Unless an end-to-end traffic matrix is available beforehand, the bandwidth requirement of the LSPs is usually unknown and has to be guessed for the first time LSPs are deployed. Later, the measured rate of the LSPs can be used as the bandwidth requirement of the LSPs.

LSP paths can be manually specified or dynamically computed. Unless offline Constraint-based Routing is used to compute the paths, manually specifying paths for LSPs is difficult. Therefore, LSPs are usually dynamically computed by an online Constraint-based Routing algorithm in the routers.

Important LSPs, such as those carrying large amount of traffic, can be given a higher priority than other LSPs. In this way, these LSPs are more likely to take the optimal paths. This will result in higher routing stability and better resource utilization from a global perspective. Multiple parallel LSPs can be configured between an ingress-egress pair. These LSPs can be placed on different physical paths, so that the traffic load from the source to the destination can be distributed more evenly. By using multiple parallel LSPs, the size of each LSP is also smaller. These LSPs can be routed more flexibly. These are the primary motivations for parallel LSPs. It is recommended that parallel LSPs be used to keep the size of each LSP below 25 Mbps Affinity, or color, can be assigned to LSPs and links to achieve some desired LSP placement. For example, if network administrators want to prevent a regional LSP from traversing routers or links outside the region, color can be used to achieve the goal. All regional links can be colored *green*, and all inter-region links can be colored *red*. Regional LSPs are constrained to take only *green* links. In this way, regional LSPs can never traverse any inter-region link. The process of assigning color to LSPs and links is again guided by administrative policy. Depending on the stability of the network, when better paths become available, network administrators may or may not want to switch LSPs to the more optimal paths. The switching of LSPs to better paths is called LSP *re optimization*. Re optimization is not always desirable because it may introduce routing instability. In the case that re optimization is allowed, it should not occur too frequently. Performing re optimization once per hour may be a good choice. As to the resilience attribute, LSPs are generally allowed to be rerouted when failure occurs along their paths. In the cases of failure, it may even be desirable to reroute LSPs regardless of their bandwidth and other constraints.

1.4 N jobs M machine Job Sequencing Technique

Routing problems in networks are the problem related to sequencing and, of late, they have been receiving increasing attention. Such problems usually occur in the areas of transportations and communication. A network problem involves the determination of a route from source city I to destination city J for there exist a number of alternative paths at various stages of the journey. The cost of journey, which may be function of distance, time or money, is different for different routes and the problem is to find the minimum cost route. The following algorithm is used to find the shortest path of the given complex network. In this technique there are N no. of jobs and M no. of machine. Each job contains constant execution time for each machine. Each job is organized such as that minimum optimal execution time is obtained. For this there is n-1 sequence is obtained by Johnson's rule and now calculates the optimum sequence.

The rest of the paper is organized as follows: In section 2, we mention related works carried out by other researchers. Proposed algorithm to find the shortest path of a given complex network is presented in section 3. In section 4, we present our results and its analysis. Finally, conclusion and future scope have been given under section 5.

2. RELATED WORK

General issues of supporting MPLS Traffic Engineering are identified and discussed in [5]. With Differentiated Services (Diffserv), packets are classified at the edge of the network. The Differentiated Services-fields (DS-fields) [12] of the packets are set accordingly. In the middle of the network, packets are buffered and scheduled in accordance to their DS-fields by Weighted Random Early Detection (WRED) and Weighted Round Robin (WRR). Important traffic such as network control traffic and traffic from premium customers will be forwarded preferably [8]. In addition to the concept of a hybrid of L2 and L3 forwarding, label distribution, and LSP setup trigger mode, the authors have proposed a framework for IP multicasting in MPLS domains. However, they did not address issues related to traffic engineering of multicasting or aggregating label assignment schemes in MPLS domains. The proposed ERM scheme eliminates most of the problems mentioned in [1,2,3] An MPLS Multicast Tree (MMT) scheme was introduced in [7] to remove multicast forwarding state in non-branching nodes by dynamically setting up LSP tunnels between upstream branching nodes and downstream branching nodes. Like ERM, MMT can dramatically reduce forwarding states. However, MMT still needs to set up and update LSPs between edge LSRs and core LSRs (if some core LSRs is branching nodes of multicast trees). As a result, the core LSRs have to support the coexistence of L2/L3 forwarding schemes. Normally LSPs are built between edge LSRs. LSPs produced by MMT may not necessarily be able to aggregate with other unicast LSPs. However, in ERM, there would be no need to set up any LSPs between edge LSRs and core LSRs, which enables ERM to aggregate both multicast and unicast traffic. Another difference between MMT and ERM is that the multicast tree is centrally calculated in MMT, while basic ERM is fully distributed, and the extended ERM (ERM2) is partially distributed. The most popular and widely used routing algorithm in MPLS networks is the shortest-path first algorithm (SPF) based on the number of hops. SPF selects the path that contains the fewest hops between the source and the destination node describe in [9, 10].

One obvious problem with SPF is that it tends to route traffic onto the same set of links until these links' resource are exhausted. This leads to concentration of traffic on certain parts of the network. In addition, SPF typically accepts less path setups into the network than some other more advanced routing algorithms. Amore intelligent routing algorithm the Minimum Interference Routing Algorithm (MIRA) proposed in [13]. The objective of MIRA is to accept as many path setups into the network as possible by using the concept of critical links. Critical links have the property that when their capacity is reduced by 1 bandwidth-unit, the maximum data flow between a given source-destination node is also reduced by 1 bandwidth-unit. The goal of MIRA is accomplished by selecting paths that contain as few critical links as possible. However MIRA suffers from two weaknesses. First, MIRA is computationally expensive.

3. PROPOSED METHODOLOGY

For MPLS traffic engineering there are so many approaches are used to control the congestion of traffic and improve the qos of the network. And for congestion control there are many control policies are used. Fair Queuing is one of them. Fair queuing is a congestion control policy where separate gateway output queues are maintained for individual end-systems on a source-destination-pair basis. When congestion occurs, packets are dropped from the longest queue. At the gateway, the processing and link resources are distributed to the end-systems on a round-robin basis. Round-robin

is an arrangement of choosing all elements in a group equally in a circular. Equal allocations of resources are provided to each source-destination pair. Here in this paper we used another algorithm for scheduling instead of Round Robin algorithm. This algorithm is based on N jobs M machine Job sequencing technique [17]. The following algorithm is used to find the optimized sequence. For this, it generates up to n-1 sequences. Sequence generation is accomplished in the following manner: Let t_{ji} where j=1, 2, 3, ……., n and i = 1, 2, 3, ……, n represent the distance having j^{th} node from the i^{th} node. In this algorithm firstly we make a weighted adjacency matrix using the given network having node and edge having some weight which denotes distance between the two nodes. Now we have divided it (N x N weighted matrix) into N X 2 sub matrix according to the given formula.

$M_{j1}^{k} = \sum_{i=1}^{k} t_{ji}$ = Constructing first column of N x 2 adjacency matrices.

$M_{j2}^{k} = \sum_{i=n+1-k}^{n} t_{ji}$ = Constructing second column of N x 2 adjacency matrices.

After it using Johnson's rule we have sequence the node so that we have obtain n-1 sequences. Then we calculate the cost of each and every sub sequence. Now estimate which sequence cost is minimum that is optimal sequence. And according to the sequence scheduler is worked. And using this technique congestion of the traffic is reduced and packet loss is negligible. The proposed algorithm is described follows:

Step 1: Begin
Step 2: Construct the N x N adjacency matrices Where N is the node of the network.
Step 3: The N x N adjacency matrices split into N x 2 sub matrices. The number of such matrices will b N - 1. Thus a network having 7 nodes then it will involve 7-1=6 sub matrices.
Step 4: Using p, where p<=N-1, auxiliary N-1 sub matrices can be defined as follows. In the K^{th} auxiliary problem.
Step 5: Set k=1, for first auxiliary problem.
$M_{j1}^{k} = \sum_{i=1}^{k} t_{ji}$ = Constructing first column of N x 2 adjacency matrices.
$M_{j2}^{k} = \sum_{i=n+1-k}^{n} t_{ji}$ = Constructing second column of N x 2 adjacency matrices.
Step 6: Apply S.M. Johnson's n-job, 2- machine algorithm to the n-job 2-machine problems established and determine S_k and store.
Step 7: Check k with p; if k < p, set k = k+1 and repeat the **step 4**; if k=p, then proceed
Step 8: Using real N x N matrix of processing distance, compute total processing distance for each of the p sequences generated.
Step 9: Select minimum total processing distance sequence as the optimal sequence. This optimal sequence is determining the shortest path of the given complex network.
Step 10: End

4. RESULT AND PERFORMANCE ANALYSIS

4.1 Simulating Parameter

For simulation topology we used two source nodes, two edge routers one is for Ingress router and other is for Egress router, two core routers and one destination node. Both source node linked with Ingress router by Duplex link. Ingress router is connected with core1 and core2 router by simplex link, core1 and core2 also connected with ingress router by simplex link.core1 and core2 linked with Egress router by simplex link and vice versa. And Egress router is connected with destination node by duplex link.

TABLE 1.a

Cir	1000000
Cbs	3000
Pir	3000
Pbs	3000
Packet size	1000
Simulation time	10.0

Queue limit	3

TABLE 1.b

Link To	Link From	Link Type	Bandwidth	Delay
N0	E0	Duplex	10mb	5ms
N1	E0	Duplex	10mb	5ms
E0	C0	Simplex	10mb	5ms
E0	C1	Simplex	10mb	5ms
C0	E1	Simplex	5mb	5ms
C1	E1	Simplex	5mb	5ms
E1	N3	Duplex	10mb	5ms

4.2 Results

Our simulation shows that on increasing the no. of packets on different nodes shows greater packet loss. Scheduling techniques used in this simulation is Round Robin.

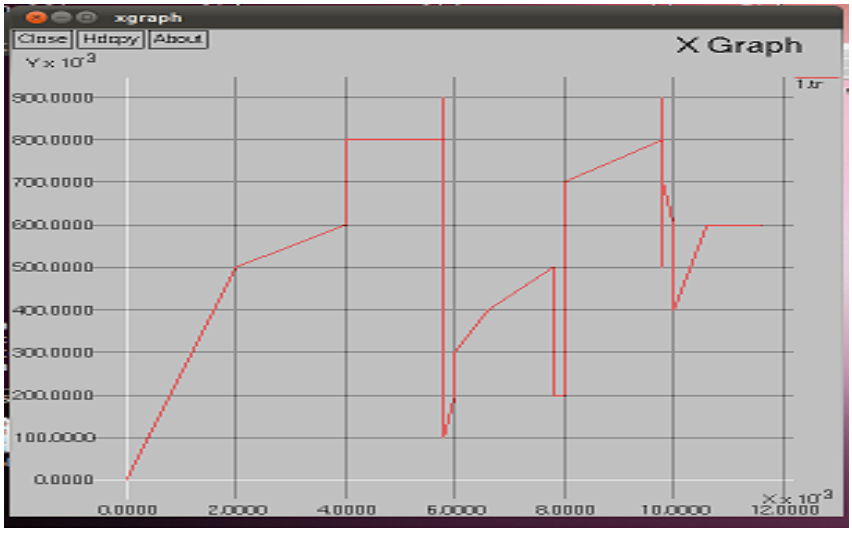

FIGURE 4.1: Lost of the application attached to agent UDP0

Our simulation with the proposed algorithm shows no packet loss has been occurred during the transmission of packets this means it provides greater efficiency than the previous scenarios.

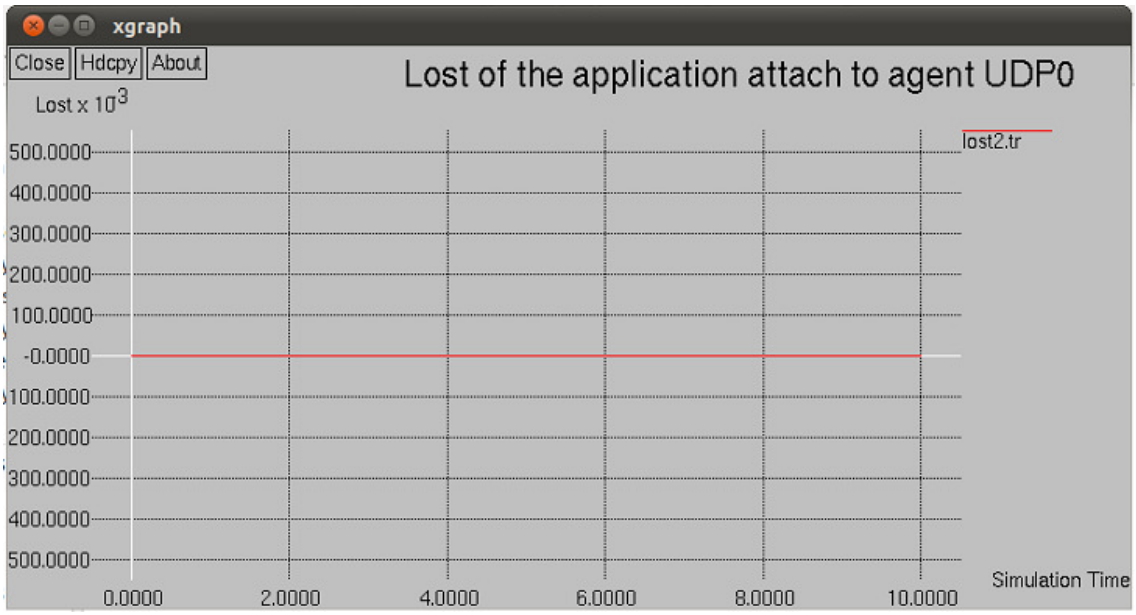

FIGURE 4.2: Lost of the application attach to agent UDP0

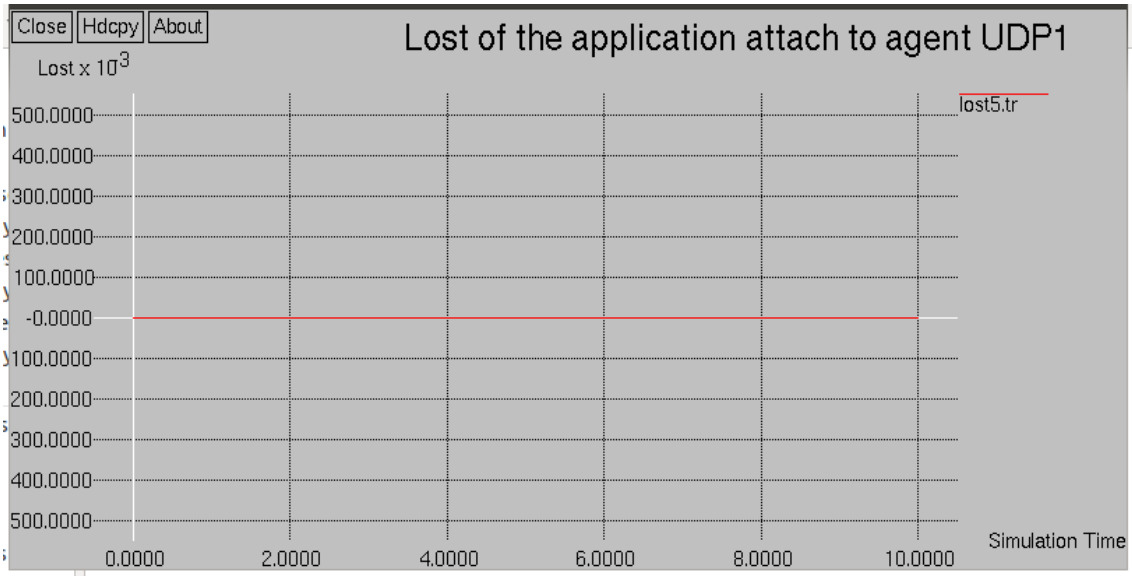

FIGURE 4.3: Lost of the application attach to agent UDP1

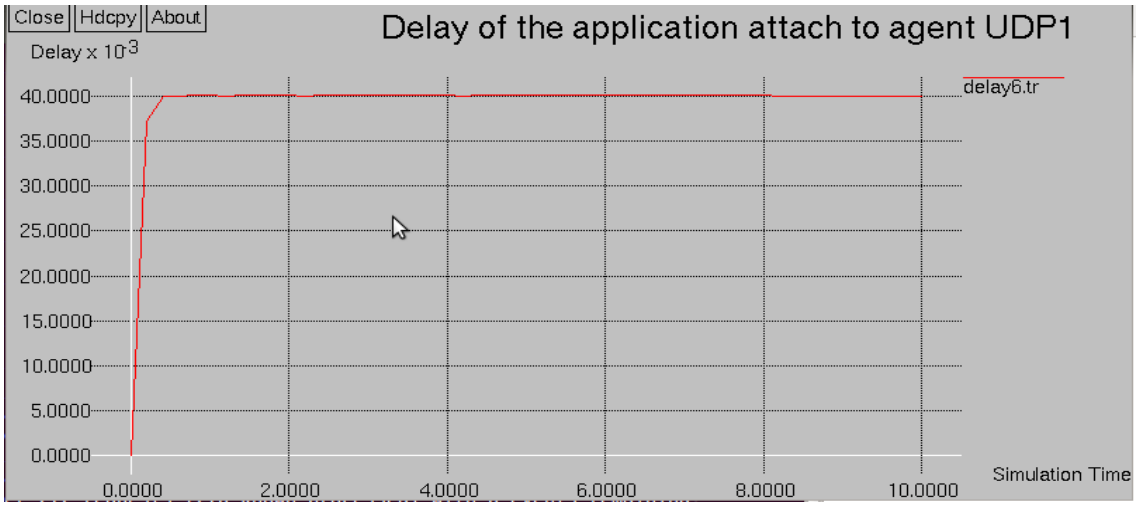

FIGURE 4.4: Delay of the application attach to agent UDP1

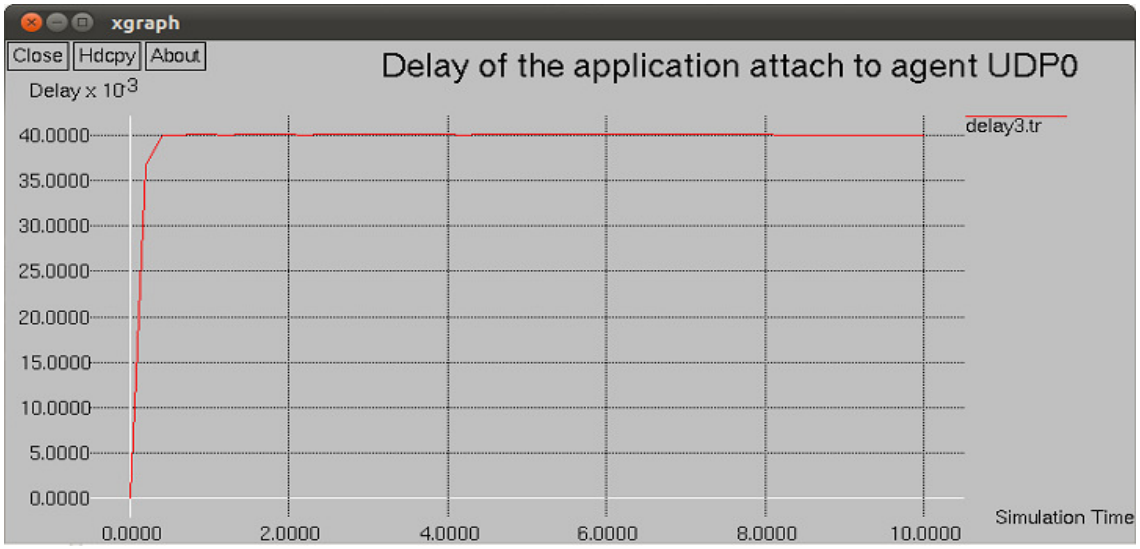

FIGURE 4.5: Delay of the application attach to agent UDP0

The figure below shows the bandwidth utilization of the current scenarios with proposed scheduling policy.

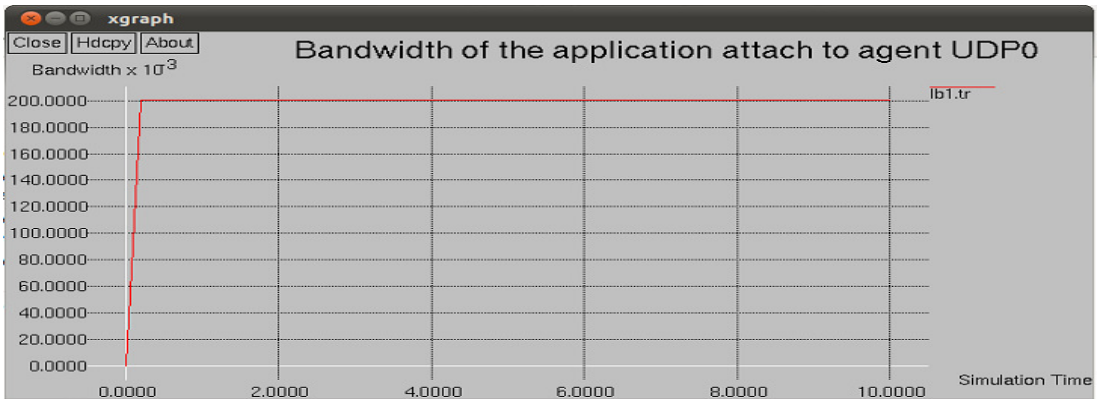

FIGURE 4.5: Bandwidth of the application attach to agent UDP0

5. CONCLUSION AND FUTURE SCOPE

The scheduling policy applied before have very large number of packet loss during the transmission of data packet. Simulation results shows that there occurs great improvement in efficiency in the network for the packet transmission, the loss of packets become nearly null for the proposed scheduling technique and the scheduling policy also maximizes the bandwidth utilization of the network. The simulation supports our scheduling policy and supports its applicability on MPLS Traffic Engineering over the Diffserv network.

In this report, many other issues are still to be resolved and need to be worked upon. Following are some suggestions to extend this work.

This topology is working on wired scenarios not in wireless scenarios. In future there can be further evaluation of our scheme for wireless scenarios or mobile MPLS. Here we work on the congestion control for the traffic engineering this algorithm can be applied for the other qos of the network ex. path optimization.

6. REFERENCES

[1]. J. Lawrence, Designing multiprotocol label switching networks, Communications Magazine, IEEE, Volume 39, Issue 7, July 2001.

[2] Luc De Ghein, MPLS Fundamentals, Cisco Systems, Cisco Press 800 East 96th Street Indianapolis, ISBN: 1-58705-197-4, 2007.

[3]. Vivek Alwayn, Advanced MPLS design and Implementation, Cisco Systems, Cisco press 201 west 103rd Street Indianapolis, 2001.

[4]. Ivan Pepelnjak , Jim Guichard, MPLS and VPN Architecture, Cisco Systems, Cisco press 201 West 103rd Street Indianapolis, March 2001

[5]. How to configure MPLS VPN over ATM using cell mode MPLS with BGP or RIPv2 on the customer site: “http://www.networkworld.com/community/node/30050 ” (last access date November 2010).

[6]. Amir Ranjbar, "CCNP ONT Official Exam Certification Guide", First Edition, Published by: Cisco Press 800 East 96th Street Indianapolis, IN 46240 USA 2007, ISBN-10: 1-58720- 176-3, ISBN-13: 978-1-58720-176-9.

[7]. DiffServ: Scalable End-to-End QoS Model on Cisco Systems site: “http://www.cisco.com/en/US/technologies/tk543/tk766/technologies_white_paper09186a00800a3e2f.html” (last access November 2010).

[8]. R. Braden, D. Clark, S. Shenker, Integrated Services in the Internet Architecture: an Overview, RFC 1633, June 1994.

[9]. Voice Over IP – Per Call Bandwidth Consumption, Cisco Systems site, Document ID: 7934, on site: "http://www.cisco.com/application/pdf/paws/7934/bwidth_consume.pdf" (last access November 2010).

[10]. H.320 Gateway to H.323 Gatekeeper Video Call Flow, Cisco Systems site, Document ID:72056."http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a0080 7 ca099.shtml#bandwidth" (last access November 2010).

[11]. Mohamed EL Hachimi,Marc-Andr´e Breton,Maria Bennani, Efficient QoS implementation for MPLS VPN, 22nd International Conference on Advanced Information Networking and Applications – Workshops, IEEE, Issue Date: 25-28 March 2008.

[12]. Fan Ya-qin, Wang Lin-zhu, Zhang Li-cui. Computational Intelligence and Design, 2008. ISCID '08. International Symposium, IEEE, Issued at Wuhan, Issued on Oct. 2008.

[13]. Fan Ya-qin, Wang Lin-zhu, Zhang Li-cui,Research for QoS of MPLS VPN based on Log-infinitely Divisible Cascades. IEEE, 2008 International Symposium on Computational Intelligence and Design, Issued Date: Oct. 2008.

[14]. C. Huang, and Vishal Sharma, "Building Reliable MPLS Networks Using a Path Protection Mechanism," IEEE Communication Magazine, Issued on Mar. 2002, pp. 156 – 162.

[15]. Antonis Nikitakis, Antonis Nikitakis. "A Multi Gigabit FPGA-based 5-tuple classification system". IEEE Communications Society at ICC, issued on 2008.

[16]. Brian Morgan, Neil Lovering "CCNP ISCW Official Exam Certification Guide", First Edition Published by: Cisco Press 800 East 96th Street Indianapolis, IN 46240 USA 2007, ISBN-13: 978-1-58720-150-9, ISBN-10: 1-58720-150-x.

[17] punit kumar singh, rakesh kumar "Path Optimization Algorithm For Network Problems Using Job Sequencing Technique", IJDPS, Issued on May 2012 vol-3.

Impact of Asymmetry of Internet Traffic for Heuristic Based Classification

Chris Richter *richter@imn.htwk-leipzig.de*
Faculty of Computer Science
HTWK Leipzig
Leipzig, 04277, Germany

Michael Finsterbusch *finster@imn.htwk-leipzig.de*
Faculty of Computer Science
HTWK Leipzig
Leipzig, 04277, Germany

Klaus Hänßgen *haenssge@imn.htwk-leipzig.de*
Faculty of Computer Science
HTWK Leipzig
Leipzig, 04277, Germany

Jean-Alexander Müller *jeanm@hft-leipzig.de*
Department of Communication and Computer Science
Hochschule für Telekommunikation Leipzig
Leipzig, 04277, Germany

Abstract

Accurate traffic classification is necessary for many administrative networking tasks like security monitoring, providing Quality of Service and network design or planning. In this paper we illustrate the accuracy of 18 different machine learning algorithms with different statistical parameter combinations. Additionally, we divide the statistical parameters into upstream and downstream to observe the influence of the protocol inherent differences of client and server behaviour for traffic classification. Our results show that this differentiation can increase the protocol detection rate and decrement the processing time.

Keywords: flow classification, Internet traffic, traffic identification.

1. INTRODUCTION

For operation, management and design of communication networks, advanced knowledge of transmitted protocols and applications as well as their behaviour are necessary. This detailed information can be provided by traffic classification. Network traffic classification, or classification of applications, is the process of identifying the type of protocols or applications which generate particular network flows. Scope of applications for traffic classification are, for example, the providing of information about future traffic evolution (trend analysis), traffic engineering, intrusion detection and prevention, content control/filtering, monitoring and lawful interception.

In general, there are four kinds of traffic classification methods. The oldest and most commonly used method is the *port based* approach. This uses the well-known port numbers of the TCP/UDP protocols assigned by the IANA. Another method used is *protocol decoding*. It is based on stateful reconstruction of sessions and application information from packet content. It identifies protocols by their characteristic protocol headers (magic numbers, incrementing counters, session identifiers, etc.), packet sequences, etc. so it avoids needing to trust in port numbers. This method is often used only for dedicated popular protocols, e.g., HTTP and mail protocols like

in Cisco's Network Based Application Recognition (NBAR) [1]. The third method is the *pattern* or *signature based* approach [2]. This method uses application specific signatures and searches for those in the protocol header and content to identify the application.

The fourth method is based on the *machine learning* approach. This method uses machine learning algorithms as used in data mining to identify applications by characteristic packet or flow statistics. The advantage of this approach is that the algorithms can be trained with real network traffic. If a protocol changed or a new protocol appeared, it is easy to repeat the training to update the protocol identifier. It can possibly be used to identify some encrypted protocols. A problem of this method is to find the proper parameters and effective machine learning algorithms.

In this paper we evaluate the impact of the network protocols asymmetry behaviour of Internet traffic for classification with a large set of machine learning algorithms and parameters. In our investigations we define asymmetry behaviour as the different behaviour of the protocols in their downstream and upstream directions. The machine learning approach has been discussed in numerous papers [3, 4, 5, 6, 7, 8], but with focus on just one algorithm. Furthermore, these approaches are mostly used for non real-time or offline network traffic analysis [3, 4, 6]. Besides the evaluation of how parameter reduction can influence the accuracy of classification, we aim to evaluate the influence of the classification runtime. These results may reveal an opportunity for using machine learning algorithms in future real-time classification.

The remainder of this paper is structured as follows: Section 2 contains a description of the experimental setup of our research, and Section 3 focus on the results of the traffic classification. In the following Section 4, we are describing the influence of parameter reduction according to the classification accuracy and the time consumption. Section 5 compares our results with the results of other studies. Finally, Section 6 provides a conclusion and the direction for future work.

2. EXPERIMENTAL SETUP

We used different network traffic traces in PCAP (packet capture library) format [9] to test and train the investigated machine learning algorithms of this study. To extract the necessary detailed protocol information and the required statistical parameters of these traces, we used our own developed tool described in [10], because available tools like GTVS (Ground Truth Verification System) [11] do not fulfil our constraints for the automatic traffic labelling. To describe the protocol characteristics, we used 40 different parameters, which are a subset of the six parameter classes: packet count, interarrival time, payload size, flow duration, bulk mode and idle mode. Some of these parameters characterize the whole flow, while the remaining parameters characterize the flow separately for upstream and downstream. This separation is done to observe the impact of the asymmetric behaviour of the network protocols for the classification. More detailed description of the 40 parameters can be found in [10].

The network traffic classification based on the protocol characteristics is done with 18 different machine learning algorithms – also called classifiers. These classifiers are provided by the WEKA software suite [12] and we treat them as black-box classifiers. The selection of these 18 classifiers is described in [10]. For the automated supervised training (Phase 1) as well as for testing (Phase 2) and protocol classification, respectively, we build a test-suite on top of WEKA.
The training data contains all statistical parameters and the associated protocol. Therefore, we can use the supervised learning approach for the machine learning algorithms. The testing data contain only the statistical parameters. During the training the classifiers generate a classifier model; this model can be used afterwards for testing in Phase 2 with different traffic. The classification accuracy of the classifiers can be validated by comparing the prediction of the classifier with the known protocol information. Due to a lack of publicly available network traces with full payload, which is necessary to evaluate the exact protocol or application of the traffic, we generated different kinds of traffic. More information to the used traffic can be found in [10].

The process of generating the classifier models is deterministic for our training data and classifiers, with the exception of the AttributeSelectedClassifier. Thus, the particular generated

classifier models are always the same for a given training set. The test results computed by the classifiers with their applied model are deterministic, too.

3. RESULTS

Table 1 contains the results of this study the filled symbols are the added results from Section 4. It shows if it is possible for an algorithm to detect a protocol with an accuracy greater than or equal to 90% with our selected parameters. Furthermore, Table 1 differentiates the results into three categories to observe the flow-direction asymmetry of the investigated network protocols:

- *full*: all 40 parameters
- *splitting*: parameters which are computed separately for the directions upstream and downstream (26 parameters)
- *no splitting*: parameters which are calculated for the whole flow (14 parameters)

WEKA Classifier Name	Bittor-rent	eDon-key	Flash	HTTP	IMAP	Oscar	POP3	RTP	SIP	SMTP
AttributeSelectedClassifier	○ Δ □	▲	○ Δ □			○ Δ □	○ Δ ■	○ Δ ■	○ Δ □	● ■
Bagging	● Δ □	▲ ■	○ Δ □	▲		○ Δ □	○ Δ □	○ Δ ■	▲ □	○ Δ ■
BayesNet	○ Δ □		○ Δ □	▲		○ Δ □	○ Δ ■	● ▲ □	○ Δ □	○ Δ □
DataNearBalancedND	○ Δ □	▲ ■	● Δ □	● ▲ □	●	○ Δ □	○ ▲ ■	○ ▲ ■	○ Δ □	● Δ □
DecisionTable	○ Δ ■		○ Δ □			▲	● ■	○ Δ ■	● ■	
FilteredClassifier	○ Δ ■	● ▲ ■	○ Δ ■	▲ □		○ Δ □		● ▲ ■	○ Δ ■	
J48	● Δ □	● ▲	● Δ □	● ▲ ■		○ Δ □	○ Δ □	○ ▲ □	○ Δ □	○ ▲ ■
J48graft	○ Δ □	● ▲ ■	○ Δ □	● ▲ ■		● Δ □	○ Δ ■	○ Δ □	○ Δ □	● ▲ ■
NaiveBayes	○ Δ □		○ Δ ■	● ■	■		● ▲ ■	● ▲ □	○ ▲ ■	●
NaiveBayesUpdateable	○ Δ □		○ Δ ■	● ■	■		● ▲ ■	● ▲ □	○ ▲ ■	●
nestedDichotomies.ND	○ ▲ ■	▲ ■	● ▲ □	● ▲ ■		● Δ ■	○ Δ ■	○ ▲ □	● ▲ ■	● Δ ■
OneR	○ Δ ■	● ▲	○ Δ □					● ▲ □	○ Δ ■	
PART	○ Δ □	▲	○ Δ □	● ▲ □		○ Δ □	○ Δ □	○ Δ □	○ Δ □	○ Δ ■
RandomCommittee	○ Δ □	● ▲ ■	○ Δ □	○ Δ □		○ Δ □	○ Δ □	○ Δ □	○ Δ □	○ Δ □
RandomForest	○ Δ □	▲ ■	○ Δ □	○ Δ □		○ Δ □	○ Δ □	○ Δ □	○ Δ □	○ Δ □
RandomSubSpace	○ Δ □	● ▲ ■	○ Δ □	○ Δ □		○ Δ □	○ Δ □	○ Δ □	○ Δ □	○ Δ □
RandomTree	○ Δ □	▲ ■	○ Δ □	● Δ □	●	● ▲ □	○ ▲ ■	○ Δ □	○ ▲ □	○ ▲ □
REPTree	● Δ □	▲	○ Δ □	▲ □		○ Δ □	○ Δ □	○ Δ ■	▲ □	○ Δ ■

TABLE 1: Protocol classification with greater than or equal to 90% accuracy (○ full, Δ no splitting, □ splitting upstream/downstream; ●▲■ additional from parameter reduction).

3.1 Classification Accuracy

It can be seen from Table 1 that not all algorithms used are suitable for protocol classification with our selected statistical parameters. The classifiers DecisionTable, nestedDichotomies.ND, OneR, NaiveBayes and NaiveBayesUpdateable have low classification accuracy over all protocols. In contrast, the classifiers RandomCommittee, RandomForest and RandomSubSpace have a high classification accuracy on all protocols, except the two protocols eDonkey and IMAP, which were classified by all algorithms with low accuracy.

Also, the results in Table 1 show that the observed protocols have different characteristics, so that some could be detected with high accuracy (Bittorrent, Flash) while others (eDonkey, IMAP) are hard to detect. Because of the specific characteristics, there are also differences in the

classification accuracy for the three parameter categories. For example, the HTTP protocol has the best classification results when using the parameters which are computed for the flow ("splitting") with differentiation between upstream and downstream. In contrast, for the protocol POP3 the classification results have a higher accuracy with the combination of all parameters ("full").

3.2 Training and Testing Effort

Table 2 contains the time needed for training and testing. These times are measured by using the HPROF [13] tool for heap and CPU profiling. We measured the CPU usage time for every algorithm and applied the fastest algorithm (REPTree with parameter category "no splitting") as reference to scale the timing results.

As we can see in Table 2, the amount of time spent for training is much higher than for testing, but this is not a problem in general. Training is done only once, while testing is done on an ongoing basis for protocol classification. All algorithms have very similar CPU time consumption, but four algorithms (BayesNet, NaiveBayes, NaiveBayesUpdateble and ND) have a significantly higher CPU time consumption. This could make these four algorithms unusable for real-time traffic classification.

WEKA Classifier Name	full		splitting		no splitting	
	Train	***Test***	***Train***	***Test***	***Train***	***Test***
AttributeSelectedClassifier	48.3	3.6	32.0	3.3	20.7	3.4
Bagging	180.1	2.1	112.5	2.0	70.6	1.9
BayesNet	62.1	74.9	37.3	46.4	21.4	25.7
DataNearBalancedND	144.2	7.7	74.2	7.4	43.3	6.5
DecisionTable	631.5	2.6	386.3	2.1	199.0	1.7
FilteredClassifier	32.2	3.3	20.7	2.5	12.5	2.3
J48	59.9	3.1	38.4	2.9	25.8	2.9
J48graft	85.8	4.0	54.7	4.2	38.1	3.2
NaiveBayes	86.9	200.7	57.1	133.3	31.3	73.7
NaiveBayesUpdateable	86.7	200.3	57.8	133.5	31.7	73.9
nestedDichotomies.ND	198.5	66.8	139.3	66.6	87.4	64.0
OneR	7.2	1.6	4.8	1.3	3.2	1.1
PART	111.5	2.9	61.9	2.6	51.3	2.5
RandomCommittee	53.3	3.5	40.9	3.0	29.8	.2.6
RandomForest	47.8	3.6	38.1	3.1	31.3	2.6
RandomSubSpace	100.9	2.9	65.7	2.4	38.6	2.3
RandomTree	6.9	1.7	4.8	1.5	4.1	1.3
REPTree	19.6	1.7	11.9	1.4	8.1	1.0

TABLE 2: Classifier time factors.

In Table 2, the time factor for training as well as for testing indicates a relation between the number of parameters and the time consumption of the classifiers. Thus, the time consumption is reduced by using fewer parameters for the protocol classification. Because of these results, we supposed a linear relation between the number of parameters and the time consumption of the classifiers. To verify this supposition, the following Section 4 includes further tests with reduced parameters for the protocol classification.

4. PARAMETER REDUCTION

Because of the test results of the previous Section 3, we decided to evaluate the influence of parameter reduction according to the time consumption of the classifiers and the accuracy of the

protocol classification. For the parameter reduction, we divided the parameters of each parameter category (“full”, “splitting” and “no splitting”) into six different parameter classes. Furthermore, we tested each classifier with all possible 63 combinations of these parameter classes – the 64th combination (all parameter classes removed) was omitted.

4.1 Classification Accuracy

The filled symbols in Table 1 represent the additional classification results of the parameter reduction. In contrast to the results of Section 3, the classification accuracy in the results were increased for all protocols and some classifiers. The improvement of eDonkey and HTTP is significant when comparing all protocols. The classification results of eDonkey show differences in the classification accuracy for the three parameter categories. The best classification results were reached with the parameters which are computed for the whole flow (“no splitting”). In addition, the classification accuracy of the classifier NaiveBayes, NaiveBayesUpdateable and nestedDichotomies.ND could be proliferated. However, Table 1 does not show which parameter combinations were suitable for the best classification results.

Despised the improvement of the classification accuracy by the parameter reduction, the classification accuracy of IMAP is still low. Only the four classifiers DataNearBalancedND, NaiveBayes, NaiveBayesUpdateable and RandomTree are able to classify this protocol with an accuracy of greater than or equal to 90%. We can see the same results on the two classifiers DecisionTable and OneR, having still the lowest classification accuracies according to all ten protocols. Even they show some improvements of the classification while using parameter reduction.

4.2 Classification of eDonkey

In fact, the classification accuracy of eDonkey reached in most cases values of only about 25%, and in some cases up to 80%. The reduction of the statistical parameters could increase the classification for all protocols, but the improvement of eDonkey was significant. Fig. 1 shows the results of the classification for eDonkey with all combinations of the six parameter classes. Every combination consists of one or more parameter classes, and every parameter class is referenced by a number (see Fig. 1). The sum of the reference numbers is assigned to those combinations that consist of more than one class.

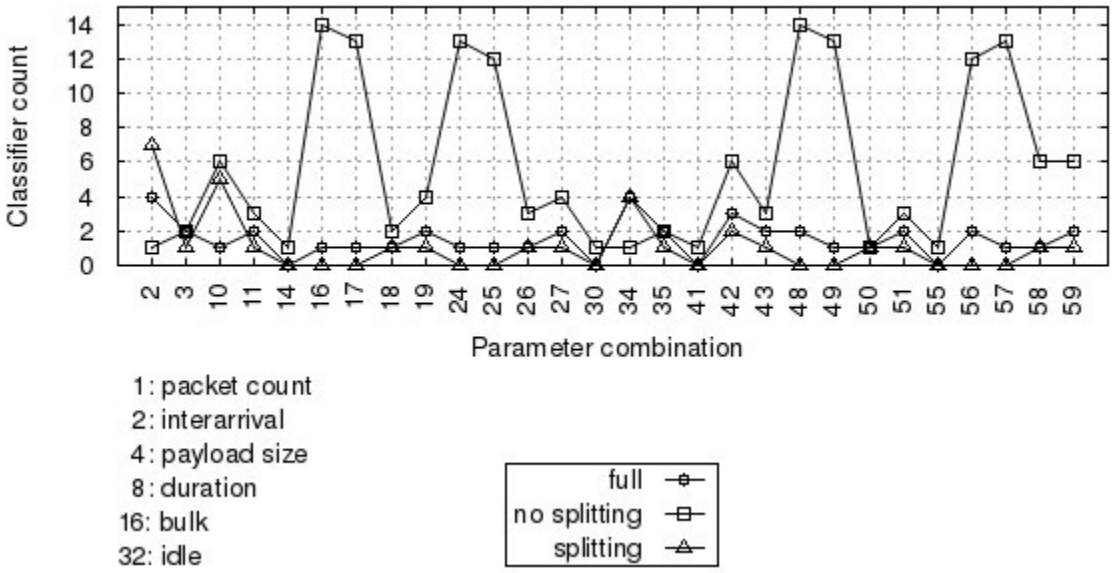

FIGURE 1: The number of classifiers that match eDonkey with a particular parameter combination with an accuracy of 90%.

Fig. 1 shows the number of classifiers that reach an accuracy of 90% or more for the different parameter combinations. We can see that some parameter combinations of the parameter category “no splitting” gain significantly more classification accuracy than the other parameter combinations. These parameter combinations are 16 (bulk), 17 (bulk + packet count), 24 (bulk, duration), 25 (bulk, duration, packet count), 48 (bulk, idle), 49 (bulk, idle, packet count), 56 (bulk, idle, duration) and 57 (bulk, idle, duration, packet count).

It is evident that the classes interarrival and payload size are not in any parameter combination that has a high accuracy. This can be explained with the nature of eDonkey's Peer-to-Peer (P2P) protocol behaviour. The eDonkey network packets differ in size because the configuration and management packets contain less data, whereas packets for data transfer can contain more data. Due to the P2P behaviour of eDonkey, many connections to peers spread over the whole world can be established. Thus, the parameter classes payload size and interarrival are not good criteria for eDonkey classification.

In contrast, the parameter class “bulk” is very important for detecting eDonkey. All parameter combinations with high accuracy contain bulk. The bulk transfer mode is a typical characteristic of data transfer protocols without application level acknowledgements. The parameter ``bulk" may also be able to increase the detection accuracy for other protocols used for data transfer like FTP, HTTP, other P2P file-sharing protocols or file-sharing integrated in instant messaging or VoIP protocols/applications.

4.3 Testing effort

As seen in Table 2, the time needed for training and testing seems to be correlated with the number of parameters used for classification. In this section, we want to have a closer look at the connection of runtime and the number of parameters. The diagrams of Fig. 2, Fig. 3 and Fig. 4 show the runtime of the different classifiers and the number of parameters used for testing. The timing results were split into three diagrams because of different scaling and different correlations between parameter count and runtime.

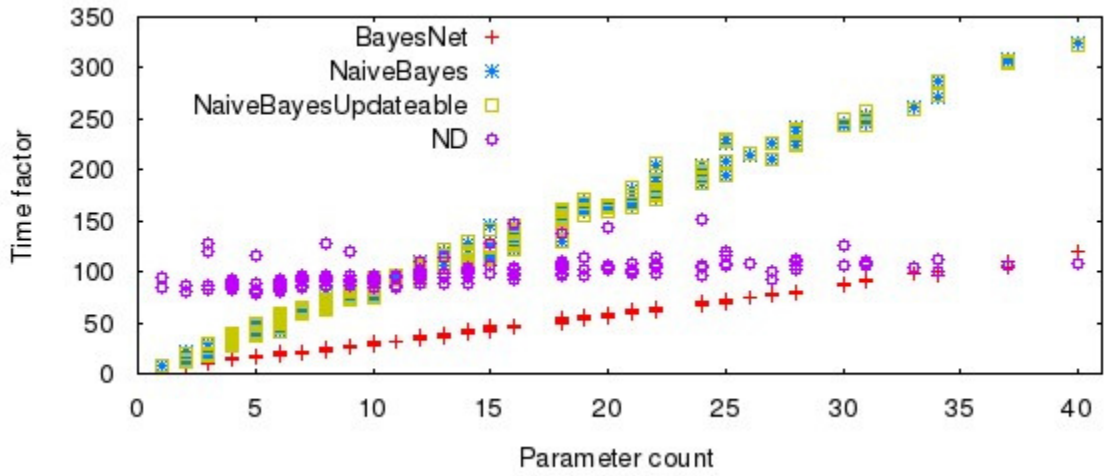

FIGURE 2: Linear time factor growth of slow classifiers.

Fig. 2 shows the runtime of the classifiers BayesNet, NaiveBayes, NaiveBayesUpdateable and ND. These classifiers are the slowest of all determined classifiers. All four classifiers rise linearly with growing parameter count. The increase of runtime for the three Bayes classifiers is significantly higher than on all other classifiers. ND has an increase similar to the classifiers in Fig. 3, but it has a huge offset. That means ND is per se very slow – independent of the parameter count.

The classifiers in Fig. 3 also increase linearly too, but they grow much slower. OneR, RandomForest and REPTree are the fastest classifiers with lowest increase. The classifiers in Fig. 4 do not show a linear increase as well. The parameter count is not the only factor for the growing runtime. The lower and upper bounds of the scatter plots in Fig. 4 are linearly increasing.

As a result, we can say that the runtime for the most classifiers is correlated to the amount of parameters. So, the reduction of parameters – with equal classification accuracy – is desirable. The Bayes classifiers are very slow and should only be used with combinations having few parameters. A prediction of the runtime of the classifiers of Fig. 4 is not possible, but the lower and upper bounds of the scatter plots give an indication. Furthermore, the ND classifier is very slow in all cases and is not suitable for real-time classification.

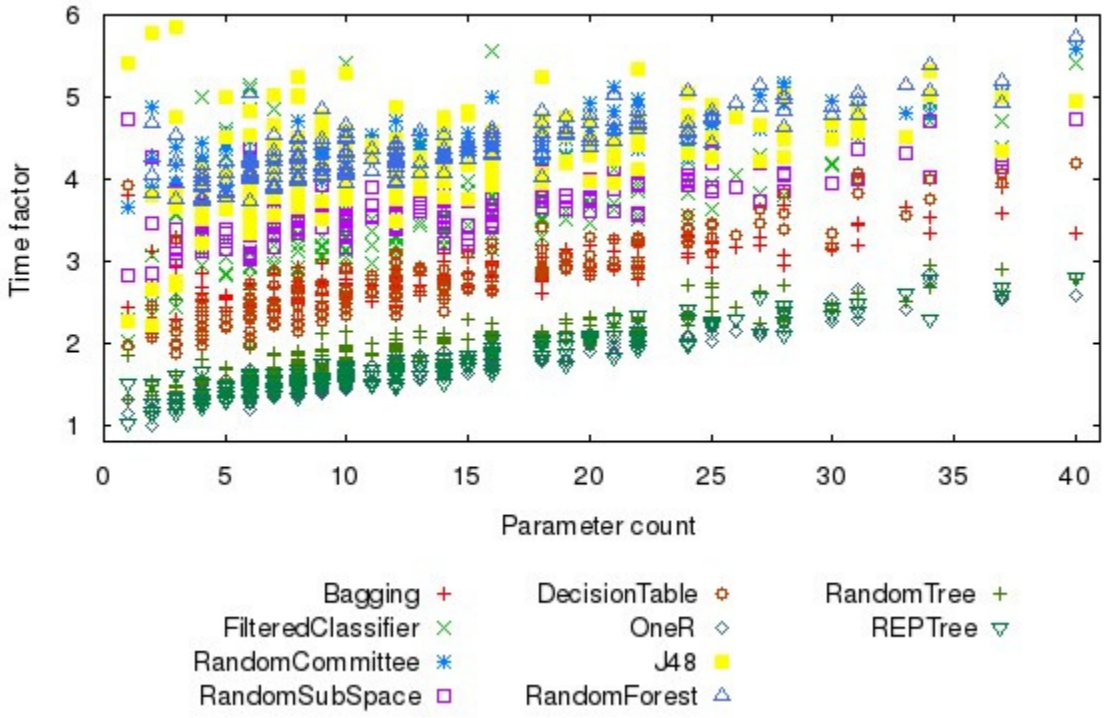

FIGURE 3: Linear time factor growth of fast classifiers.

5. Related Work

As described in the introduction, the machine learning approach has been discussed in numerous papers [3, 4, 5, 6, 7, 8] before. In this section we want to compare our findings with the results of previous papers and expose the differences. First, all previous papers used only one algorithm to classify the traffic. In some papers, however, different methods to improve the algorithms were used additionally [7, 8].

A big issue in all related work is the data pool of network traffic. There are no up to date full payload internet traffic traces available. All public available traces are truncated after the transport layer header. So, a responsible estimation of the included payload and the used upper layer protocols is not possible. Therefore, port numbers are often used to classify the truncated packets [6]. For this paper we classified and verified all traffic flows by hand to give the machine learning algorithms reliable information for the learning phase. In addition, the knowledge of the available protocols and applications is important for validating the results.

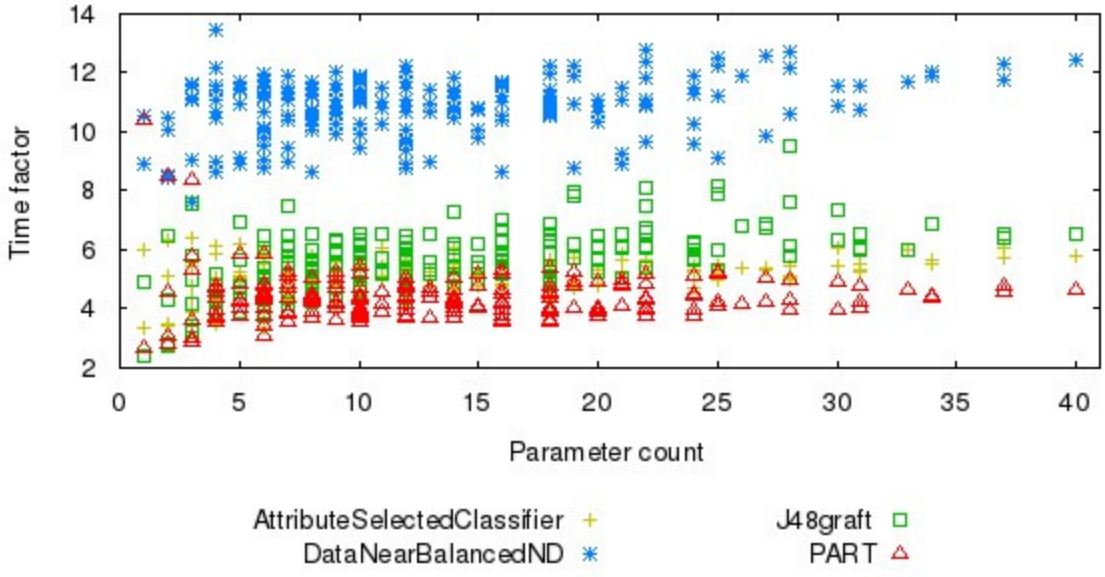

FIGURE 4: Classifiers without determinable time factor behaviour.

Another key issue is the choice of network protocols to investigate. All studies – this study included – are using only a small amount of about ten protocols or applications. For real use the 50 most used protocols should be observed. This issue is mostly due to the lack of available network traces. But this can sophisticate the results. If we use only very heterogeneous protocols, the classification is, on the one hand, easier and the machine learning algorithms will work more responsibly; on the other hand, the *false positive* rate is much lower. This effect can be seen in the results of Table 1. The protocol IMAP, for example, was falsely predicted as another protocol. The same behaviour can be seen in paper [3]. In this paper the detection rate (*true positive*) is 80% or more. However, the protocol POP has a classification accuracy of 0%, because it is very similar to NNTP and SMTP which results in false classification (false positive). Because of the small number of investigated protocols – in all studies – the effect of false positives is not considered sufficiently.

Besides the 18 machine learning algorithms, we investigated 40 statistical parameters to determine their influence on traffic classification. We only used statistical parameters, whereas other studies have also used parameters like IP addresses or port numbers [4, 8], which in effect reduces the machine learning approach to absurdity. As discussed in Section 4, the kind of statistical parameters and the number of parameters is important for the classification accuracy. This result can also be seen in [3, 6, 7]. However, this is dependent on the observed protocol and the used machine learning algorithm – a key result which cannot be read out of the other studies [3, 4, 5, 6, 7, 8].

6. CONCLUSION AND FURTHER WORK

In this paper, we have determined the impact of network protocol asymmetry according to the classification accuracy of network traffic. We used 63 combinations of six statistical parameter classes in three parameter categories consisting of up to 40 parameters. The three categories split the statistical parameters into parameters for the whole flow without differentiation of upstream and downstream, parameters that differentiate between upstream and downstream with regard to the asymmetry of some protocols, and the third category containing all 40 parameters.

Our test results show that the differentiation of the traffic can increase the classification accuracy if the parameters can expose the asymmetry of protocols like HTTP, which has an asymmetric

behaviour. This asymmetry can be lost if the statistical parameters are only computed for the whole flow. Furthermore, the reduction of parameters can increase the classification accuracy. This is significant for the eDonkey protocol. Removing the parameter classes “interarrival” and “payload size” enhanced the classification accuracy of 14 of the 18 classifiers to at least 90% when the parameter “bulk” was used. This indicates that the parameter ``bulk" can be used to detect asymmetric bulk data transfer.

Additionally, the parameter reduction can decrease the runtime of the classifiers. Most classifiers have a linearly increasing runtime with reference to the parameter count. This is important for real-time traffic classification. The classifier ND showed a low classification accuracy and a high runtime, so it can be dismissed for network traffic classification. The Bayes classifiers should only be used for a very small number of parameters, because their runtimes increase very fast with increasing parameter count. Besides, the NaiveBayes and NaiveBayesUpdateable classifier show a similar behaviour and classification accuracy, and therefore, in future work it is sufficient to investigate only one of these two algorithms.

6.1 Further Work

In future we want to investigate if it is possible to train the machine learning algorithms for protocol classes such as e-mail, bulk data transfer, P2P, interactive, gaming or multimedia to enable the classification of unknown protocols belonging to such a class.

The network traffic used in this study does not reflect network traffic in the Internet. Because other studies showed that the accuracy of the classification results can vary by testing network traffic from other locations [7, 14, 15, 16], we have to repeat our investigation with other network traffic to evaluate our results.

To obtain better classification results, we have to study the best suitable algorithms in detail to adapt these generic algorithms for traffic classification. Another result of this study may be an answer to the question of why some classifiers are more suitable for traffic classification than others.

Finally, to enhance the accuracy of traffic classification, we have to find other parameters. New parameters should take the payload characteristics into account. Above all, parameters which characterise payload properties of the network protocols can enhance the classification accuracy of encrypted protocols.

6.2 Acknowledgements

We thank the anonymous reviewers for their insightful comments. This work is supported by the European Regional Development Fund (ERDF) and the Free State of Saxony.

7. REFERENCES

1. “Network Based Application Recognition (NBAR),” Cisco® , Oct. 2011, http://www.cisco.com/en/US/products/ps6616/ products ios protocol group home.html.

2. “Application Layer Packet Classifier for Linux,” Oct. 2011, http://l7-filter.sourceforge.net.

3. L. Bernaille et al., “Traffic classification on the fly,” *SIGCOMM Comput. Commun. Rev.*, vol. 36, April 2006. [Online]. Available: http://doi.acm.org/10.1145/1129582.1129589

4. H. Jiang et al., “Lightweight application classification for network management,” in *Proceedings of the 2007 SIGCOMM workshop on Internet network management*, ser. INM ’07. New York, NY, USA: ACM, 2007, pp. 299–304. [Online]. Available: http://doi.acm.org/10.1145/1321753.1321771

5. T. T. Nguyen and G. Armitage, “A survey of techniques for internet traffic classification using machine learning,” *IEEE In Communications Surveys & Tutorials*, vol. 10, no. 4, pp. 56–76, 2008.

6. S. Zander, T. Nguyen, and G. Armitage, “Automated traffic classification and application identification using machine learning,” in *Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference* on, Nov. 2005, pp. 250–257.

7. P. Piskac and J. Novotny, “Using of time characteristics in data flow for traffic classification,” ser. AIMS’11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 173–176. [Online]. Available: http://dl.acm.org/citation.cfm?id=2022216.2022243

8. A. W. Moore and D. Zuev, “Internet traffic classification using bayesian analysis techniques,” in *Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, ser. SIGMETRICS ’05. New York, NY, USA: ACM, 2005, pp. 50–60. [Online]. Available: http://doi.acm.org/10.1145/1064212.1064220

9. “TCPDUMP & LibPCAP,” http://www.tcpdump.org.

10. M. Finsterbusch, C. Richter, and J.-A. M¨uller, “Parameter Estimation for Heuristic Based Internet Traffic Classification,” in *ICIMP 2012: The Seventh International Conference on Internet Monitoring and Protection*, IARIA, Ed. Stuttgart, Germany: IARIA, 2012, ISBN: 978-1-61208-201-1.

11. M. Canini, W. Li, and A. W. Moore, “GTVS: boosting the collection of application traffic ground truth,” University of Cambridge, Tech. Rep. UCAM-CL-TR-748, 2009. [Online]. Available: http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-748.pdf

12. M. Hall et al., “The WEKA Data Mining Software: An Update,” *SIGKDD Explorations*, vol. 11, no. 1, 2009.

13. K. O’Hair, “HPROF: A Heap/CPU Profiling Tool in J2SE 5.0.”

14. A. W. Moore, M. L. Crogan, and D. Zuev, “Discriminators for use in flow-based classification,” Queen Mary University of London, Tech. Rep., 2005.

15. A. Finamore et al., “Kiss: Stochastic packet inspection classifier for udp traffic,” *Networking, IEEE/ACM Transactions* on, vol. 18, no. 5, pp. 1505 –1515, 2010.

16. C. Rotsos et al., “Probabilistic graphical models for semi-supervised traffic classification,” in *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference*, ser. IWCMC ’10. New York, NY, USA: ACM, 2010, pp. 752–757. [Online]. Available: http://doi.acm.org/10.1145/1815396.1815569

Enhance the Security and Performance of IP over Ethernet Networks by Reduction the Naming System Design

Waleed Kh. Alzubaidi *waleed@ieee.org*
Information Technology Department
University Tun Abdul Razak
Selangor,46150, Malaysia

Dr. Longzheng Cai *charles_cai@unitar.my*
University Unitar International
Selangor,46150, Malaysia

Shaymaa A. Alyawer *sha_amh@yahoo.com*
Computer Science Department
Baghdad College
Baghdad, 645, Iraq

Abstract

In this research, we investigate the weak link between two protocols, IP protocol and Ethernet protocol. IP over Ethernet network has become the major network used by Internet. In this network, still the data link layer performance and security problems not adequately addressed yet. The findings of this research lead us to propose a modification, by making a reduction on current naming architecture to improve the network performance and security. The proposed architecture will be evaluated by a theoretical analysis.

Keywords: IP, MAC address, Ethernet, ARP, Security, Performance.

1. INTRODUCTION

Despite the Internet is widely adopted and success, still its architecture is far from ideal. The Open System Interconnect (OSI) model divides network functions into layers. Services are provided from lower layers to upper layers without the knowledge of each other. This model provides simplicity in working and flexibility in developing, and allows the changes been made in specific layers without the need of changes in other layers. For instance, when started to developing 802.11b wireless networks, changes were made only in data link layer and physical layer. On the other side, this model allows the performance problems and security flaws of lower layers affect upper layers. For example, the performance effect on resolving layer 2 addresses (MAC address) will delay or prevent connection setup in the network layer. The security weaknesses in data link layer may compromise the whole communication [14]. While there are several ways to enhance the performance and security in upper layers, still the problems in data link layer have not been addressed adequately yet. Although network devices like switches and bridges have provided some performance enhancements and security features, problems are still there. The design of the current naming architecture in IP over Ethernet networks that use to deliver the data within Local Area Network was not enhanced since it was founded. The current design is not ideal [1], and need be revised to make it more effective and secure. In this research, we will focus on improve performance and security in the data link layer of IP over Ethernet networks.

Data link layer is the last layer before the data is converted to physical signal. The network traffic at this layer considers a complete traffic. It contains data that want to be sent and the control information which include destination and source addresses (IP, MAC), protocol type and port

number, etc. That means any problem in data link layer may lead to more sophisticated problems in performance and security issues in the upper layer.

We found several performance and security problems in data link layer were generated due to using the current naming architecture in IP over Ethernet networks. This architecture used to accomplished data transfer inside the local area networks, and it consider one of the requirements for binding network layer address and data link layer address.

First, current naming architecture uses two different address forms, which introduces an overhead by constantly mapping between IP and MAC address.

Second, the mapping process is considering an extra process added to current data transmission procedure. This considers a delay issue when data been stored and not sent until the address resolving processes complete.

Third, Performance problems and security issues in data link layer may not be able to reduce or avoid it in the upper layers. The performance problem in data link layer has a direct effect on the whole communication, and upper layers cannot provide a solution. Therefore, it is better solve these problems from the base of the ISO model. However, a more efficient naming architecture may able to use one addressing type after initiation network stack, so no more constantly mapping needed. We can use only IP address to identify a host, and use it as a destination address in sending Ethernet frame to the target node. This new naming architecture may require a change in current naming architecture in Ethernet networks. Still network devices like bridges and switches need to maintain a table for IP and their ports.

In this study, we research on the weak link between Ethernet and IP protocol, and proposed a compatible-backward modification on current naming architecture to secure the data link layer in IP over Ethernet networks. Moreover, improve the performance by reducing on the address mapping in the data transmission process. This new proposed solution will be evaluated by theoretical analysis.

2. LITERATURE REVIEW AND BACKGROUND

2.1 ARP Overview

In the local area network, the Address Resolution Protocol (ARP) is used to map IP address to MAC address. To construct and transmit Ethernet frame in IP over Ethernet networks, destination MAC address should be obtained by source machine by using a destination IP address. This task performed by ARP protocol by broadcast a request for mapping IP to MAC address and store reply in a memory space called ARP cache table. ARP works as follows: an application attempts to send data to an IP address of a machine. IP packet will be created by the network stack, and then encapsulated inside Ethernet Frame. For transmission this frame, it needs the destination MAC address. Therefore, the network stack checks the IP in the ARP cache table to find the destination MAC address. If it is not there, then broadcast ARP request on the network. Each machine in the network will examine the ARP request and check if they own the requested IP. The machine that owns this IP will create ARP reply containing their MAC address. Then, send unicast reply to the originator of this request. The originator will use this address in destination MAC address field to complete the frame and transmit it. ARP is a simple statelessness protocol. Also ARP considers a layer 3 protocol. It does not design to have any security aspects to bind IP and MAC address.

The basic idea is that the router is configured to reply to ARP requests on behalf of the hosts on the other side of the router. When the original host receives the reply, it is not aware that the MAC address it is receiving does not belong to the destination host, but to the interface of the router on the current network. Gratuitous ARP is unsolicited ARP messages sent by hosts, directed to their own IP addresses. Hosts commonly use this type of messages when joining a network with a

dynamically assigned IP address. These hosts use gratuitous ARP to confirm that the newly assigned IP address is not currently in use by another host in the network.

Moreover, a host broadcast a gratuitous ARP when it is initializing its IP stack. The gratuitous ARP is an ARP request message to verify there is no conflict IP address. By gratuitous ARP, the host asks for Layer 2 address of its own IP address[9].As shown in Figure 1.a.

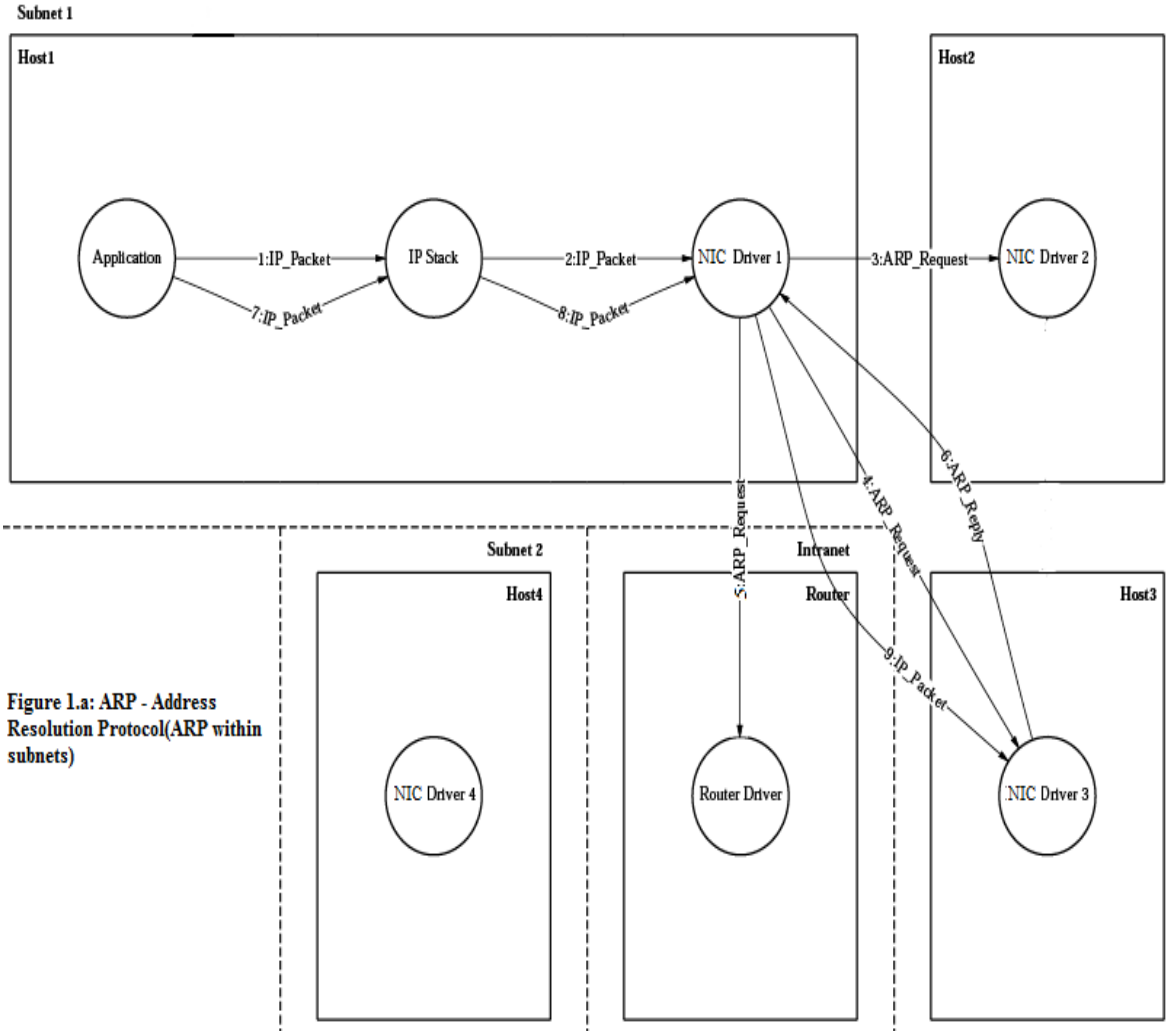

Figure 1.a: ARP - Address Resolution Protocol(ARP within subnets)

3. Analysis of Data Link Layer Performance

Current naming architecture in IP over Ethernet networks has many performance problems. To start Layer 2 communication, system will create IP packet at Layer 3 and it will be encapsulated at Layer 2 inside Ethernet frame. Before the system sends this frame it needs the destination hardware address (MAC address). System will do ARP request and queue the information until resolving the destination address. In the next step and if the remote host is reachable will return ARP reply message with its MAC address. Local system will be including this MAC address in the Ethernet frame header then send it. The remote host will receive and de-capsulate the frame and proceed the datagram to the upper layer. To explain the performance problems in data link layer and make it more clear, let suppose we have two computers. Local computer want to check remote computer is alive or not. Figure 2.1 show the procedure of checking remote host with ping program. Ping program is used for checking, and it will generate ICMP packet at layer 3 for send it to the remote computer. The system needs the destination hardware address to send this

frame. The system will do ARP request and at the same time queue the ICMP packet until can resolve remote hardware address. Remote computer will receive the request message and return ARP reply with its MAC address. After receiving ARP reply, local computer will encapsulate the ICMP packet inside Ethernet frame and include the destination MAC address in the header of the frame. Remote computer will receive the frame and return ICMP echo reply. Finally, Local computer after receive ICMP reply can determine the remote computer is alive. There are many problems in this mechanism due to use current naming architecture.

First, delay actual Data that want to be send while system queuing it until can be resolve the destination hardware address. The mapping messages susceptible to lose,

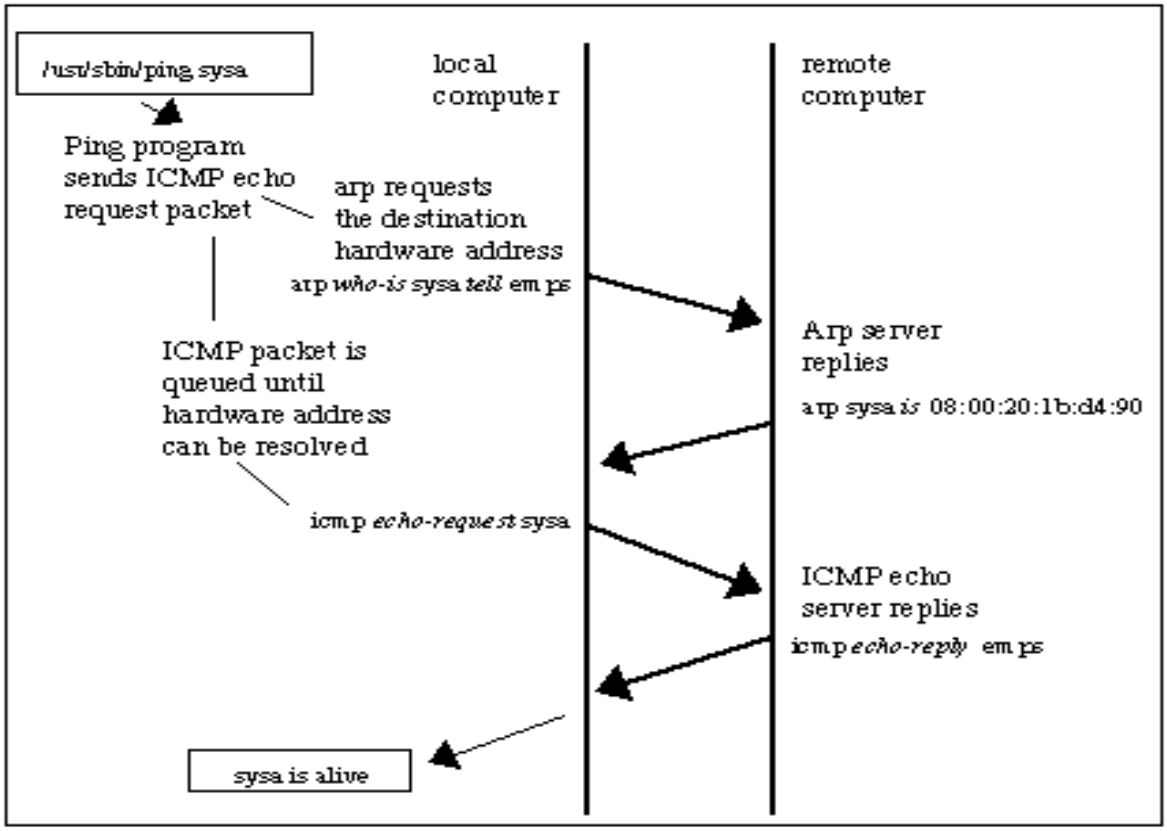

FIGURE 2: Ping program procedure

damage or delay due to media performance or attacks threads. The mapping process considers an extra process that is making data transmission process is heavier and more complex.

Second, in the mechanism of checking whether the remote computer is live or not, it is need four steps to determine. While already we understand the remote is live from the second step, because if the remote is not living so who reply ARP message. Our thesis focuses on these problems, to reduce queue time and to send data directly. Moreover, reduce the procedure to determine remote target is alive or not in two steps instead four. These problems emphasis our theory that the problem in the design of the naming architecture in IP over Ethernet networks.

4. Ethernet and IP protocol Link Security Problems

Data traffic in the data link layer considers a complete traffic. It is including the information that wants to be sent and the control information. The control information include destination and

source IP and MAC address, port number, protocol type, etc. Therefore, controlling data link layer traffic will compromise the whole communication.

Ethernet protocol composes from Media Access Control operations and Ethernet frame structure. Ethernet was not design specifically to work with IP protocol and it is different. But the needed lead to made IP protocol work over Ethernet. The framework of IP protocol work over Ethernet is not fully compatible. This caused a weak link between Ethernet and IP protocol. This weak link appears clearly in data link layer where the joint between IP protocol and Ethernet, by mapping between IP and MAC addresses and encapsulate IP packet inside Ethernet frame. However, the weak link generates gaps that may use to exploit the communication in data link layer.

During these days TCP/IP has become the major protocol that used because the Internet depends on it widely. And Due to vary usage of Internet It is revealing shortcomings. TCP/IP protocol use IP address to identify the host, while Ethernet identify the host by MAC address. Data transfer need to use Ethernet address to deliver Ethernet frame from a network machine to another in the same subnet. The routing in the data link layer depends on upper protocol to deliver frame to the right node. Moreover, Ethernet protocol level needs a resolving mechanism to map the carry protocol address to Ethernet address. Therefore, one of the requirements to made IP work over Ethernet is founding a binding mechanism to bind between IP and MAC address. For instant in IP protocol, Ethernet protocol level need the destination IP address to resolve destination MAC address. Address Resolution Protocol (ARP) was founded to map the IP address to Ethernet address (MAC Address) in the local area network. When a host wants to know the MAC address of the destination IP address, it broadcasts an ARP request including the destination IP address of the target on the network. The nodes that match the IP address in the message return ARP reply with its MAC address. Each node in the network builds a table, called ARP cache, used to store the mapping IP addresses to MAC addresses.

In this section we will discuss attacks behavior and see there is some types depend on mapping process and other depend on MAC address and there is another with different criteria. It is beneficial to understand how layer 2 attacks works and what the dependency points that needs by attack to enable. So we can manipulate these points and can determine what types of attacks will affect by the proposed solution. These dependency points can be targeted in finding an appropriate solution. In our thesis we will focus on attacks that depend mainly on ARP protocol to redirect the traffic. Furthermore, the attacks that depend on the MAC address. The most known Layer 2 attacks are Denial of Service (DoS), Man In The Middle (MITM), MAC spoofing, ARP poisoning, MAC flooding, and port stealing. In Layer 2 based Denial of Service DoS attack, the attacker updates the ARP caches for the network hosts by sending nonexistent MAC addresses. The attacker uses a Layer 2 based DoS attacks to disable layer 2 network connection to the victim and then uses its pair IP and MAC addresses. Each network interface card in the network is supposed to have a globally unique MAC address[16]. It is well known physically burn and non-changeable. However, now it can be easily to change inside the operating system to enabling MAC cloning attack. The hijacking attack that is Layer 2-based, an attacker impersonate of the connection between two network hosts.

Man In The Middle (MITM) [2] is an attack that redirects the traffic between any two nodes in layer 2, like between host and a router. That possible, because ARP is a stateless protocol, and cannot verify the origin of the messages. Each time a host gets an ARP reply, and even it does not send an ARP request for this reply, it will updates its ARP cache table with this ARP reply [4]. The process of updating a target host's ARP cache table with a fake entry is referred to as poisoning. Attacker sends a fake ARP reply message with IP address of host B and the MAC address of the attacker to host A. Additionally, the attacker sends a fake ARP reply with IP address of host A and the MAC address of the attacker to host B. The traffic between the two hosts A and B pass through the attacker machine allowing sniffing.

Layer 2 has another attacks types, these attacks targeted switch forwarding table technic. Ethernet switch depends on CAM table to store the node’s MAC address and the corresponding

physical switch port. Normally this memory table has limited space. In the flooding attack, the attacker floods the network switch with MAC addresses using fake ARP frames to fill and overflow the CAM table. Then, the Ethernet switch starts broadcasting the traffic without switching to the right port similar to hub mode.

Port stealing is one of these attacks that depend on the switch technic to gain access to layer 2 traffic. The port stealing attack uses the mechanism of the switches in how binding MAC addresses to physical switch ports. When a switch receives Ethernet frame from a port, it binds the port number with a source MAC address. The attacker, first, floods the switch with fake ARP frames including the target host's MAC address as the source address and the attacker MAC address as the destination address in ARP reply frame. Since the target host also sends normal frames traffic, with consideration a race condition. The switch receive frames from two different ports with the same source MAC address and continuously amendment the binding of the MAC address to the physical port in CAM table. If the attacker is faster in sending frames, the frames that intend for the target host will send to the attacker's switch port instead to the target host. Attacker steals the target host's port so the traffic going through it first, and then to the target host. The attacker then will send an ARP request. The attacker request for the target hosts' MAC addresses in the ARP message.

The attacker wills stops sending fake ARP request frame during waiting for the ARP reply. The receiving an ARP reply means that the target hosts' port in the switch has been restored to the normal binding. After receiving the ARP reply, the attacker will forward the frame to the target host. Whole process will repeats by the attacker for each new frames [5] [6] [7].

There are many ways to mitigate these types of attacks. Some of the systems are monitor ARP cache table updates. And some reject unsolicited ARP reply message from updating its ARP table.

Some solutions depend on the switch to mitigate layer 2 attacks. Port security is one of the options in the switch that used to binds a physical port in the switch to MAC address. MAC flooding and cloning attacks are preventing by Port security option. Still ARP spoofing is possible and not prevent by port security [3]. Port security validate source MAC address in the frame header, whilst there is an additional source MAC field in the data payload inside ARP frames, and clients use this field to advertise their caches [10].

5. PROBLEM STATEMENT

This study have been reveals many weak points in current naming architecture in IP over Ethernet networks. Some of it related to ARP that is one of the most vulnerability points in Layer 2 and create an overhead, and some related to MAC address. The weak points hereinafter:

1. There are no security aspects for ARP protocol while mapping IP to MAC addresses as ARP RFC [10].
2. Delay actual Data in the Ethernet frame while waiting for ARP process to resolve destination MAC address. It will be more real-time if done without call sub-procedures. See Figures 5.1.a, b.
3. Using Address Resolution Protocol is susceptible network devices to various attacks in layer 2, which lead to sophisticated attacks to upper layers.
4. ARP introduces an overhead on operating system by continually sending and receiving ARP messages to map IP to MAC addresses and maintaining ARP cache table.
5. Employing ARP in current naming architecture will be generating extra noise in the network within the same collision domain. That exhausted the network resources.
6. IP protocol is not fully compatible work over Ethernet protocol, this cause a weak link that lead to many performance and security flaws, appear clearly in data link layer where IP joint with Ethernet, when encapsulate IP packet inside Ethernet frame, and resolving IP to its MAC address.

6. RELATED WORKS

In this section, we will provide an overview of secure ARP methods and new architecture that have been proposed to solve Layer 2 problem.

6.1 Cryptographic Architectures

S-ARP [19] Proposed to address the issues of ARP spoofing. It suggested messages encryptions as the basic strategic to tackle the issues of ARP spoofing. It is backward-compatible and extension to the Address Resolution Protocol, which depends on a public key cryptography to authenticate ARP replies message. To implement this solution in a LAN, every host to be secured should be modified to use S-ARP instead of ARP. Additionally, there must be a certification authority, which is called the AKD, that is contacted to obtain the public key of a host so that replies can be authenticated by The backwards-compatible design allows hosts without the middleware to function, but at the risk of being vulnerable to ARP attacks. Verifying the appended signature, The AKD also distributes its clock value so that the other hosts can synchronize to it. This is necessary to prevent replay attacks that could be performed to spoof a host that is down (or being DoSed).As a proof of concept, the system was implemented on Linux. To make the solution compatible with dynamic IP assignments, a modification of DHCP called S-DHCP is proposed. A drawback of this scheme is still employs ARP in broadcast, which will generates extra noise in the network within the same collision domain. Additionally, employ AKD server make a single point of failure in the network. If the AKD is down, a host cannot verify ARP packets that are sent by a previously unknown host (i.e., the sender's public key is not in the receiver's key ring). Even if the AKD is working properly, an attacker can impersonate a host that goes down by cloning the MAC address of the host (but only until the cache entry of the host being impersonated, in the host being attacked, expires). One of the most crucial problem with this approach is the processing time imposed to encrypt, decrypt and to send extra messages getting the public key or getting the host verified, all these extra time processing is burdening the current ARP protocol not including the current broadcast behavior.

TARP[13] 2005, Implements the security by distributing centrally issued secure IP,MAC address mapping attestations (called tickets) through existing ARP messages. These tickets are centrally generated and signed by a Local Ticket Agent (LTA), and contain an expiration time. Hosts attach these tickets to ARP replies so that the receiver can verify the validity of the address association. Versions for statically and dynamically address assigned networks also are described. TARP is backward-compatible with ARP protocol, but it is susceptible to replay attacks during a small period of time. Moreover, still have a single point of failure by relies on LTA server if it downs then the system down. The authors implemented TARP for Linux, as a combination of a kernel module and a user space daemon.

IEEE 802.1X [15] protocol is one of IEEE Standard for port-based Network Access Control (PNAC). This standard provides an authentication mechanism at layer 2 by use a modular topology to devices want to attach to network (LAN or WLAN). A user that want to accessing the network makes a request to a gateway that at the same time play the authenticator role that controls network accessing and redirect the requests to an authentication server. In addition, the authenticator and authentication server are in the same system, as is often with 802.11b wireless access points. The authentication servers may include RADIUS and LDAP.EAP allows methods of authentication like PEAP, MD5, TLS, TTLS and use optional keying material. Once this takes place, the authenticator forwards user credentials to the authentication server. The server send accept or reject message, along with user configuration data such as Virtual LAN number. Also EAPOL protocol was redrafted to use with IEEE 802.1AR and IEEE 802.1AE (MACSec) in 802.1X-2010.

The main drawback it is a non-Independent network that need additional device, and cause single point of failure. Moreover, Man-In-The-Middle and session hijacking attacks is possible. EAPOL send a Logoff frames by the 802.1X supplicant through the network in clear, and contain data for the credential exchange for initially authenticated the client. Therefore it easy to be

spoofed, and can be enable DoS attack on both wired and wireless LANs. In EAPOL-Logoff attack, constantly sends fake EAPOL-Logoff frames from the malicious third node that has access to the medium that the authenticator is attached to.

6.2 Middleware architectures

By using middleware architectures to detecting ARP attacks they will not change or extend ARP protocol. Watch the local ARP cache for changes. Analyze ARP packets. Actively validate mappings. Normally monitor a suspicious ARP behavior like was a request sent to a given reply, Invalid MAC addresses in reply and whether ARP packet breaks current mappings.

Dynamic ARP Inspection [18] some high-end Cisco switches has this feature. It is allows the switch to drop ARP packets with invalid *IP, MAC* address bindings.

To be able to detect which ARP packets have invalid bindings, the switch uses a local pairing table built using a feature called *DHCP snooping*. This scheme promises to be a very effective solution to the problem of ARP attacks, but thorough tests need to be performed to confirm if in fact it is able to prevent all types of ARP attacks. One main disadvantage of this solution is the high cost of switches that have this feature available. Additionally, depending on the setup of the DHCP server and the network, it might not be possible to validate some ARP packets on all switches in the VLAN [11].

Ebtables[12] is a Linux technique used to create programmable bridging and switching devices to perform Ethernet frame filtering, among other things. It has been suggested that Ebtables can be used to implement ARP attack prevention mechanisms, but the efficacy of such method has not been studied. The main drawback of this approach is that this solution would only filter malicious ARP messages that attempt to pass through the Linux box, while other areas of the network remain unprotected. Additionally, Ebtables rules to prevent ARP attacks are not widely available, and the task would have to be left to the administrator, who can easily make mistakes when programming the bridge or the switch.

Anticap [24] is a kernel patch for various UNIX-based operating systems that aims at preventing ARP poisoning attacks by rejecting ARP updates that contain a MAC address different from the current table entry for that IP address. This solution works in static environments, but does not work in dynamic (DHCP-enabled) networks, with no security if mapping not yet in cache, pure kernel mode processing, and is available for a limited number of operating systems (Linux 2.2/2.4, FreeBSD 4.6, NetBSD 1.5).

6.3 Operating System Architectures

Static ARP cache entries add a static entry to local ARP cache is simple and effective way to prevent ARP attacks because static entries cannot manipulate through ARP spoofing. It is good for individuals to secure ones gateway. This solution has two disadvantages: First, it does not work in dynamic environments to use DHCP (Dynamic Host Configuration Protocol).Second; it does not scale well, as it would be need more efforts for the network administrator to deploy and update these tables throughout the network because once new or changed hosts will affects all hosts.

Furthermore, some operating systems (such as Windows 2000/XP) may accept dynamic ARP replies and updates for static entries [21].

MAC spoofing attacks can be detected by sending an **Inverse ARP (InARP)** [22] request for a MAC address. The response can be used to determine if a computer is performing cloning [20] (if and only if the computer being cloned has not been DoSed or shutdown). This is a very limited solution as it only detects this type of ARP attacks.

Operating System Security Behavior In general, every operating system has different network stack and behavior, each network interface card has different drivers with their own behavior,

even the same operating system may have different behavior depending on network stack version, driver, and firmware version. In Linux kernel 2.4 does not react to unsolicited replies but Inserts mappings from requests into cache.

Some operating systems like Solaris only accept ARP updates after timeout period [20]. This makes it harder for the attacker to poison the cache, but not impossible. When this type of mechanism is used, an attacker can poison the cache as long as the attacker's ARP reply arrives before the reply from the legitimate host, or by sending a forged ICMP echo request that appears to come from one of the two victims [20].In Windows no inbuilt ARP security and Registry settings affect ARP behavior [17].

7. OBJECTIVES OF THE STUDY

In this study, the goal is to improve the security and performance of the networks by studying the link compatibility between Ethernet and IP protocols. Ethernet was not founded to work with a specific Layer 3 protocol. Also, IP protocol was not designed to work with a specific Layer 2 protocol. This makes IP protocol and Ethernet protocol relation is not fully compatible, and results to many performance and security problems. Resolving IP to MAC address and encapsulation IP packet inside Ethernet frame are the requirements to link between IP and Ethernet protocol. This clearly appears in data link layer, which is the link between the two protocols.

Five main objectives have been identified that lead to a logical progression through the thesis:

1. Improve network security especially Layer 2 security: Most serious attacks in layer 2 depend on ARP protocol to redirect data traffic. By reducing the use of ARP protocol for resolving IP address in data transfer process, and cancelling the use of MAC address in delivery data process in layer 2, we will gain a better security state as well as more reliable communication.
2. Minimize operating system resources usage: ARP functions and cache table building and maintaining are responsibilities of the operating system. By stopping and removing ARP functions (request and reply) and stopping building and maintaining the cache table from operating system duties, we can reduce the computational time in operating system procedures to transmit network information, and reduce memory space consuming that used to store cache table.
3. Minimize network resources usage: ARP request message is based on broadcasting and its reply is based on unicast. By reducing this part of traffic, network will be more silent. Moreover, many other protocols will effects with this new proposal. For Instance, ICMP protocol will proceed without resolving process that leads to enhance in protocol mechanism. Another example, DHCP protocol will still work without the need to change and without enhancing on its performance. There are still many other protocols need to determine the level of effects. This will achieve by an empirical and monitor its procedures under the new circumstances and conclude its behavior. By applying that we save network resources by reduce part of traffic and enhance protocol mechanism.
4. Make a modification on the design of the Ethernet frame header which will use IP address instead MAC address: current naming architecture using MAC address which consists from 48-bits in the Ethernet header, while IP address is consisting from 32-bits. Therefore, utilize IP address instead MAC address makes no problem with the size of address field in the frame header. First, this will achieve by build a converting form procedure. Second monitoring what other cases will effects due these changes like the broadcast and multicast traffic.
5. Evaluate operating system and network functioning due to the architecture changes: We will use our knowledge gained in previous tasks to formally evaluate the new architecture against the current state.

8. THE METHODOLOGY

Our methodology will be focus on make a reduction on Data Link Layer to obtain a higher performance and security in IP over Ethernet Networks. Some literatures recommended our vision to solve Layer 2 problems but they consider it hard to implement [23].In this section we will explain our hypothesis and modification required to achieve our objectives. The hypothesis can be described as follow:

- After a device establish its IP and subnet addresses, it is satisfactory to have only IP address of the destination device to send the Ethernet frame for that device. In this case, there is no need to use MAC address, and there is no need for an ARP cache table.
- A more efficient protocol should be able to use only one of the addresses after an initial setup. A network device may identify itself when it is first connected to a subnet work to establish a network address and use that address without constant mappings. This type of protocol requires changes in Layer 2 architecture.

We follow two factors in methodology to achieve our objectives:

First, reduce the processes of the Address Resolution Protocol. To make these changes on current naming architecture in IP over Ethernet networks, it needs to cancel the use of Address Resolution Protocol ARP from data transmission procedure. This can be achieved by using one address form for both Layer 2 and Layer 3.Inthe current procedure when network device want to send Ethernet frame to another node in the LAN, the destination IP address is known, but destination MAC address is not. At this point the system will call ARP process to resolve destination MAC address, as shown in Figure 5.1.a. After canceling the use of ARP from unicast traffic the procedure will be optimized to be as shown in figure 5.1.b.

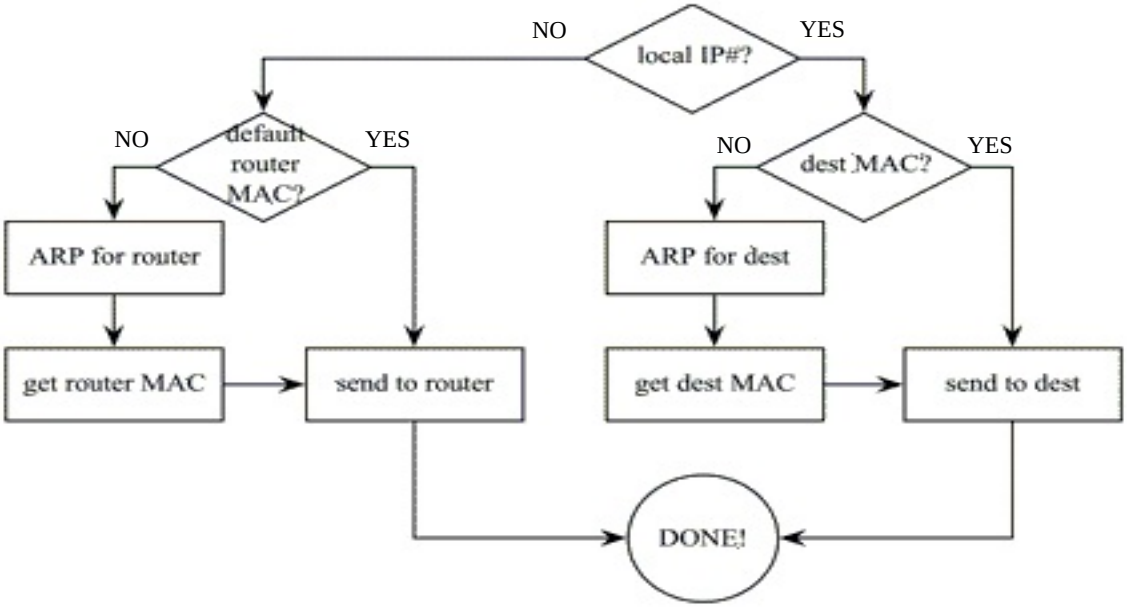

FIGURE5.1.a: Current Procedure for Transmission one Frame process

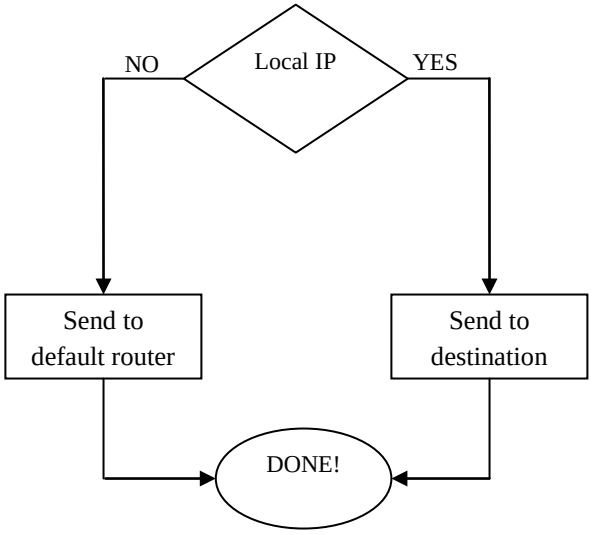

FIGURE5.1.b: Transmission process after apply the Hypothesis

Second, cancel the use of MAC address in the Ethernet frame header and use IP address from the packet (Layer 3 address). The packet in Layer 3 contains the source and destination IP address. While in Layer 2 the Ethernet frame header should fill with destination and source MAC address, see the Figure 5.2.a.

The proposed modification is to change the architecture to use IP address as a destination and source address in the Ethernet frame header instead using MAC address. See the Figure5.2.b.

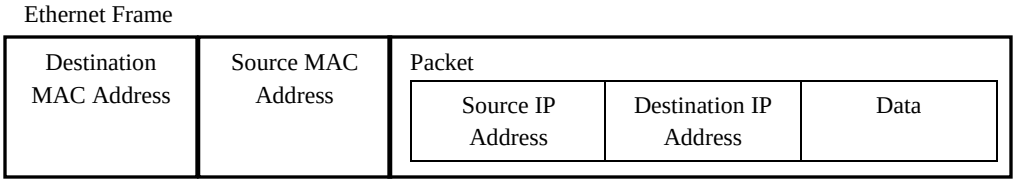

FIGURE5.2.a: Ethernet II frame

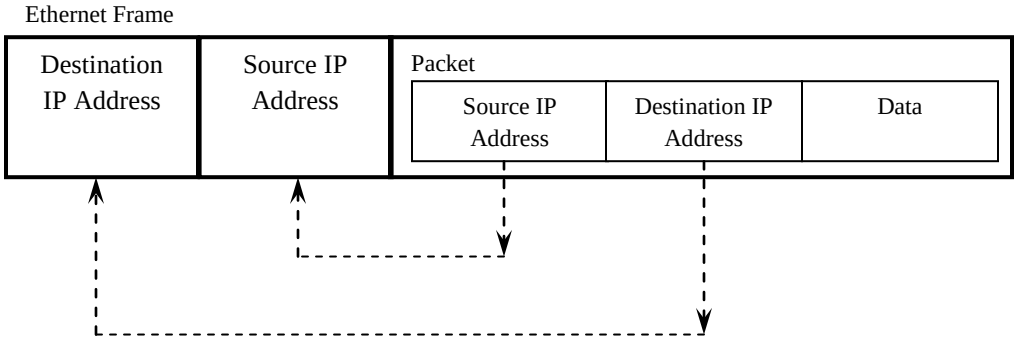

FIGURE 5.2.b: New Proposed Ethernet frame

The IP address will be used as a flat address for both Layer 2 and Layer3 and will represent in both Packet and Ethernet frame header. When received this frame by the destination node, the system will exam the address in the arrived frame header with Layer 3 address instead Layer 2 address, to check whether the frame is for this device or not. By following the above procedure in sending and receiving frame we could obtain:

- We will no need for use ARP request or reply.
- No more need for ARP cache table that take a time to process and memory space to maintain it.
- ARP processes (Request and Reply) will be optional after applying our hypothesis for use if need to inform about MAC address.
- Gratuitous ARP will still in use to detect conflict IP in Local Area Network.

From above descriptions there will be an influence significant on others protocols from all above layers. For example, ICMP protocol mechanism will optimize to be more efficient with two steps to achieve the goal instead current scheme with four steps. See Figure 5.3.

8.1 Study Phases

We can abbreviate our study passes three phases as follow:

1. **Phase 1**: Primarily analysis by collecting the result from different performance and security scenarios for the current scheme.
2. **Phase 2**: In this phase, after proposal evaluation we will start to explain the new circumstances for the proposed scheme there will be the explanation how the network components (devices and protocols) work under the proposed circumstances.

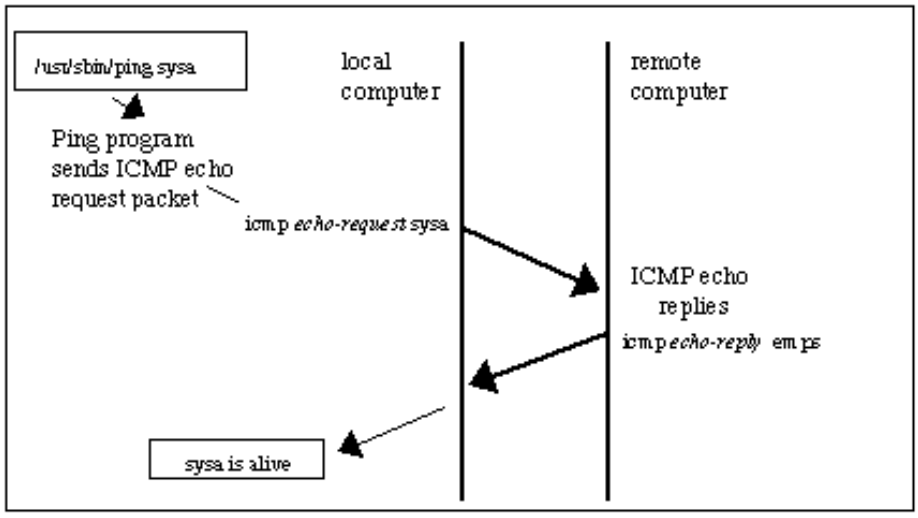

Figure 5.3: ICMP Enhancement

Main researching issues will be in:

- Router and Ethernet switch.
- Network Address Translation (NAT).
- Security issues, the attack that may still exist.
- Tunnel protocols.
- MAC unique.
- Internet Control Message protocol (ICMP).
- Traffic types: Broadcast and Multicast.
- Identity and authentication.
- Dynamic Host Configuration Protocol (DHCP).

3. **Phase 3**: Evaluating network functioning and operating system due to the architecture changes.

8.2Adoption

The literature was presented solutions that proposed extra devices like secure servers and protocols to attach to the current scheme. The complex, unforeseen problems, more administrator efforts and cost, all these factors prevent any propose solution from widely adoption. Processing the problem from the base will offer most advantages with mostly standardize. Here will illustrate how our proposal is promising to be widely adopted, and explain how network devices and protocols work under our proposal circumstances:

1. DHCP: DHCP is Layer 3 protocol, used to automatically assign IP address to LAN nodes. DHCP has no role with Data transmission. Since we are focusing on increase the performance and security for Data transmission in Layer 2 only, DHCP will still work without change. Host will still identify itself with its own MAC address to IP address.

2. Gateway Router: In the current scheme, when the network device desire to connect with external destination in WAN side, it will use gateway router MAC address in sending frame. In our proposed scheme, the destination address field in the Ethernet frame header will fill with gateway router IP address. Delivered frame will be test with router IP address, if it is equal then de-capsulate and proceed to the network layer to test the destination IP address in the packet header. The IP in the destination address field may equal to the gateway router IP address, which mean this deliver packet is send to the gateway itself, else the packet need to reroute.

3. Ethernet Switch: Ethernet switch build a table, called Content Addressable Memory (CAM) tables. It stores the binding MAC addresses to physical switch port numbers. Ethernet switch forward arrived frame to appropriate physical port depend on CAM table entries. At the same time switch look for the source address in the Ethernet frame header, and update the CAM table with this MAC address and the physical port that came from. With our propose modifications on the address in the frame header, switch will continue store this address, without recognize whether it change or it is MAC or other address. This means Ethernet switch will still work normally and no need for changes.

4. Point-to-Point connection within LAN: Network station needs IP address to attach to the network. To start a connection with another station within LAN, it needs the destination IP address of remote station, to fill the destination address field in the packet header with this address. ARP protocol is used to resolve the destination IP address to MAC address, and to fill the destination address field in Ethernet frame header with this destination MAC address. In this study we propose to use destination IP address as the address that fill with destination address field in the Ethernet frame header instead the destination MAC address. This will reduce the time and process required to send the frame, which means faster sending frame without need queue the frame until resolve hardware address No need for use ARP protocol, and no more need to build and maintain cache table.

9. COMPARATIVE EVALUATION

Most of the studies in the security and performance problems of IP over Ethernet naming architecture are focusing on ARP-based solutions architectures. While the problem is still there, that because the mapping process was founded due to the need for resolving mechanism between two different naming systems, IP and MAC address. So it is not the fault of ARP protocol. The main issues may introduce by the design of the architecture.

Whereas, our proposed modifications in the naming architecture depending on revising the origin naming architecture design, and make a reduction with replaceable parts.

ARP-based solutions depend on proposing and attach a new scheme to existence architecture. That will led to increasing the complexity degree and breaking the standards. Moreover, it may create an additional unforeseen performance problems and security vulnerabilities. Building a new architecture needs constructing everything from beginning and that will need more costs and efforts.

Whereas, in our proposed architecture we avoid to break the standard and increasing the complexity at the same time that will be achieved by avoiding propose a new architecture. Instead of that we revised the origin design and reduce the using of two addresses to using one flat address, which is the IP address from layer three. Same IP derived from layer 3 which is used in the packet that wants to send. This IP address will use in the Ethernet frame header instead the MAC address. In result, we are using same name space in IP over Ethernet naming architecture to guide the data travel through the network.

In our proposed architecture, we cancel the use of ARP protocol which will reduce the resources that need to accomplish the connection. Moreover, saving the network bandwidth by removing the heaviest bandwidth part, which it is the broadcast bandwidth that generated by ARP request messages and unicast ARP reply messages. This will provide a more reliability and increase the availability in the network connections.

Whereas, the ARP-based solution depend on existence of ARP protocol as essential to provide a protection. This will not provide any performance enhancement. On the contrary, this will increase the complexity. At the same time it may be provide a limited protection solution. Even that will break the standard and will not easy to adopt.

Still the solutions that ARP-based architecture is using MAC address as Ethernet address in layer 2. That was deployed in the destination and source hardware address in Ethernet frame header. Which need to use layer 3 IP address to resolve the destination layer 2 MAC address. Whereas, our proposed architecture no need any more for using MAC address in layer 2. Due to using IP address in the header of Ethernet frame, means the architecture will depend on the available destination IP address to accomplishing the frame transmission instead using the MAC address. That will save the efforts to bring and resolve the destination address. It is worth to mention here, that the current resolving mechanism faces many possibilities to attacks or drop or collision that will directly effects on the performance for the data queued and waiting for transmit.

In current naming architecture using ARP cache table is necessary to keep the resolved addresses for a specific period of time. This table is susceptible to attacks to poison its entries. leading to redirect the traffic toward third party point like ARP cache table overflow attack. Moreover, ARP cache table are requiring maintaining by the operating system and updating its contents. That will make the system always busy. Further, it is taking a memory space that is need also to reserve and maintain by the operating system.

Whereas, in our architecture, we proposed to cancel the use of ARP cache table due to no need for the resolution process with using one address in the naming system. That will return many advantages. For instance, avoiding the ARP cache table attacks, and save the operating system resources. Moreover, no need to update the ARP cache entries and remove the expire one. Further, canceling the resolution process will lead to cancel the reservation memory space and save the network node resources.

Most of the provided solutions are focusing on increasing the security features in IP over Ethernet naming architecture, which make the protocols heavy and more complex. Most of the data link layer protocols were not designed with or for security issues. In the literature, authentication servers, encryption algorithms and ticketing system are proposed to add to the current architecture, which make them not independent and need one more extra step to achieve the goal of the transferring process. Moreover, these solutions create a single point of failure.

On the contrary, our architecture was targeted the easy adoption goal. That was achieved by reduction the design making a significant optimization in the current naming architecture.

10. DISCUSSION

The proposed architecture will eliminate the ARP spoofing attacks in the network. That will be by preventing update the ARP cache table with unverified entries. By using one flat address in the IP over Ethernet naming architecture, that will make the system without needing for mapping process by ARP protocol. Which it is the main point to enable layer 2 attacks by redirect the traffic to the third party. In this study we proposed a modification for the naming architecture without ARP cache table. That will eliminate the possibilities for the attackers to poison the cache table with fake information. Moreover, it will provide the ability to send the data in real time without queuing the layer 2 ready data. Depending on IP address in both layer 2 and layer 3 encapsulations. Another advantage it possible to eliminate the overflow attack of the cache table, duo to there is no more use for ARP cache table. Our new method avoids breaking the standard by bringing a novel mechanism and elements to attach to the current network architecture. In result, the complexity degree will not increase and keep the current scheme simple and clear. We made a modification on current scheme by using IP address as the only one flat address for naming system in IP over Ethernet networks. That will provide easy steps to adoption like such scheme in current network architecture. Like such scheme need only an update for operating system to add the proposed modifications to meet with the needs. The most stubborn attacks in IP over Ethernet networks are Man-In-The-Middle, and this attack are prevented by the new proposed scheme. This attack depends on poisoning to redirect layer 2 traffic toward third party. Another layer 2 attack is DoS attack, by poison the cache table of the victim with nonexistence IP/MAC pair for another machine in the network, which means it can't reach for this destination.

11. CONCLUSIONS

We made an investigation on the performance and security problems in Data Link Layer, and found the problem in the link between IP protocol and Ethernet protocol. These two protocols are not fully compatible and not designed to work specifically with each other. The current design of the naming architecture may take the main issue in the problem, while we found weak principles in using MAC address. This link needs to be enhanced to improve the performance and security in the data link layer. We proposed to make a reduction on the naming architecture design. In this architecture, we utilize Layer 3 address as a flat address for both Layer 2 and Layer 3 instead of using fixed media access control (MAC) addresses. Moreover, this architecture reduces the role of ARP in the unicast data traffic.

11.1 Future Research

The presented IP over Ethernet naming architecture with the new prosed concepts is a significant step toward enhancing the performance and security of the network naming architecture. During this study we observed many new field may further researches.

The impact of the proposed architecture on the others protocols in upper layers may study further. Especially the secure protocol such HTTPS, SSL and SSH.

The security in the data link layer may need to more explore. Various attacks with different techniques in layer 2, may research to investigate which is disabled and which is effects and who still not effected.

Our scope in this study includes IP protocol working with Ethernet technology environments. Further research may work on IP protocol with different layer 2 technologies. For instance, ATM protocol with fiber optic technologies. Further, frame relay and other layer 2 protocols.

12. REFERENCES

[1] Craig A. Shue, Minaxi Gupta, An Internet without the Internet protocol, Computer Networks 2010 54 (2010) 3232–3245, http://dx.doi.org/10.1016/j.comnet.2010.06.009.

[2] NathNayak, G., GhoshSamaddar, S., “Different Flavours of Man-In-The-Middle Attack, Consequences and Feasible Solutions”, Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference.

[3] S.J. Prowell, R. Kraus, and M. Borkin, “Seven Deadliest Network Attacks”, Syngress, 2010.

[4] Bashir, M. S., “ARP Cache Poisoning with Ettercap" August 2003Available at http://www.giac.org/practical/GSEC/Mohammad Bashir GSEC.pdf.

[5] S.Vidya, R.Bhaskaran, “A Subnet Based Intrusion Detection Scheme for Tracking down the Origin of Man-In-The-Middle Attack”, IJCSI International Journal of Computer Science Issues, Vol.8, Issue 5, September 2011, ISSN(Online): 1694-0814, pp-173-179.

[6] S.Vidya, N.Gowri, R.Bhaskaran, “ARP traffic and Network Vulnerability”, in proceedings of INDIACOM-2011, conducted by BVICAM, New Delhi, India, page – 619 and in CD.

[7] Hayriye C. Altunbasak, “Layer 2 Security Inter-Layering In Networks,” Thesis dissertation, Georgia Institute of Technology, Dec. 2006.

[8] Xiangning HOU, Zhiping JIANG and Xinli TIAN. The detection and prevention for ARP Spoofing based on Snort. In 2010 International Conference on Computer Application and System Modeling (ICCASM 2010).

[9] Behrouz A. Forouzan, “TCP/IP Protocol Suite”, Fourth Edition, Tata McGraw Hill, pp. 220-223, 2010.

[10] Plummer, D. C., “Ethernet Address Resolution Protocol: Or converting network protocol addresses to 48.bit Ethernet address for transmission on Ethernet hardware." IETF RFC 826, November 1982.

[11] C. Schluting. Configure your Catalyst for a more secure layer 2, Jan. 2005. http://www.enterprisenetworkingplanet.com/netsecur/article.php/3462211.

[12] B.D.Schuymer.ebtables: Ethernet bridge tables,Mar.2006.http://ebtables.sourceforge.net.

[13] W. Lootah, W. Enck, and P. McDaniel. TARP: Ticket-based address resolution protocol. In *Proceedings of the 21st Annual Computer Security Applications Conference (ACSAC ’05)*, Dec. 2005.

[14] TJ O’Connor, “Detecting and Responding to Data Link Layer Attacks”, SANS Institute InfoSec Reading Room, Oct 13, 2010, http://www.sans.org/reading_room/whitepapers/detection/detecting-responding-data-link-layer-attacks_33513, 2010.

[15] 802.1x-2004, http://www.ieee802.org/1/pages/802.1x-2004.html.

[16] Sanjeev Kumar, Orifiel Gomez, “Denial of Service due to direct and Indirect ARP storm attacks in LAN environment”, Journal of Information Security, 2010, 1, pp. 88-94, doi:10.4236/jis.2010.12010 Published online October 2010 (http://www.SciRP.org/journal/jis).

[17] Microsoft Windows 2008 TCP/IP Protocols and Services Technical Reference, Thomas Lee and Joseph Davies, Chapter 3: Adress Resolution Protocol (ARP).

[18] C. Schluting. Configure your Catalyst for a more secure layer 2, Jan. 2005. http://www.enterprisenetworkingplanet.com/netsecur/article.php/3462211.

[19] D. Bruschi, A. Ornaghi, and E. Rosti. S-ARP: A secure address resolution protocol. In *Proceedings of the 19th AnnualComputer Security Applications Conference (ACSAC ’03)*,Dec. 2003.

[20] S. Whalen. An introduction to ARP spoofing.2600: The Hacker Quarterly, 18(3), Fall 2001 http://www.node99.org/projects/arpspoof/arpspoof.pdf.

[21] Static ARP more dynamic than you might think on, http://www.chrismc.de, last access 15/8/2011.

[22] T. Bradley, C. Brown, and A. Malis. “Inverse address resolution protocol”, Sept. 1998. RFC 2390.

[23] Altunbasak, H., Krasser, S., Owen, H., Sokol, J., Grimminger, J.,andHuth, H.-P., “Addressing the weak link between Layer 2 and Layer 3 in the Internet architecture," in Proc. of the 29th Annual IEEE Conference on Local Computer Networks (LCN), (Tampa, Florida), November 2004.

[24] M. Barnaba. anticap. http://www.antifork.org/viewcvs/trunk/anticap, August/2011.

[25] Cisco Systems. *Configuring Dynamic ARP Inspection*, chapter 39, pages 39:1–39:22. 2010. Catalyst 6500 Series Switch Cisco IOS Sofware Configuration Guide, Release 12.2SX.

INSTRUCTIONS TO CONTRIBUTORS

The International Journal of Computer Networks (IJCN) is an archival, bimonthly journal committed to the timely publications of peer-reviewed and original papers that advance the state-of-the-art and practical applications of computer networks. It provides a publication vehicle for complete coverage of all topics of interest to network professionals and brings to its readers the latest and most important findings in computer networks.

To build its International reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJCN.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Started with Volume 5, 2013, IJCN aims to appear with more focused issues. Besides normal publications, IJCN intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

We are open to contributions, proposals for any topic as well as for editors and reviewers. We understand that it is through the effort of volunteers that CSC Journals continues to grow and flourish.

IJCN LIST OF TOPICS

The realm of International Journal of Computer Networks (IJCN) extends, but not limited, to the following:

- Algorithms, Systems and Applications
- ATM Networks
- Cellular Networks
- Congestion and Flow Control
- Delay Tolerant Networks
- Information Theory
- Metropolitan Area Networks
- Mobile Computing
- Multicast and Broadcast Networks
- Network Architectures and Protocols
- Network Modeling and Performance Analysis Network
- Network Security and Privacy
- Optical Networks
- Personal Area Networks
- Telecommunication Networks
- Ubiquitous Computing
- Wide Area Networks
- Wireless Mesh Networks
- Ad-hoc Wireless Networks
- Body Sensor Networks
- Cognitive Radio Networks
- Cooperative Networks
- Fault Tolerant Networks
- Local Area Networks
- MIMO Networks
- Mobile Satellite Networks
- Multimedia Networks
- Network Coding
- Network Operation and Management
- Network Services and Applications
- Peer-to-Peer Networks
- Switching and Routing
- Trust Worth Computing
- Web-based Services
- Wireless Local Area Networks
- Wireless Sensor Networks

CALL FOR PAPERS

Volume: 5 - **Issue:** 3

i. Submission Deadline : July 30, 2013 **ii. Author Notification:** September 15, 2013

iii. Issue Publication: October 2013

CONTACT INFORMATION

Computer Science Journals Sdn BhD
B-5-8 Plaza Mont Kiara, Mont Kiara
50480, Kuala Lumpur, MALAYSIA
Phone: 006 03 6207 1607
006 03 2782 6991

Fax: 006 03 6207 1697

Email: cscpress@cscjournals.org

CSC PUBLISHERS © 2012
COMPUTER SCIENCE JOURNALS SDN BHD
B-5-8 PLAZA MONT KIARA
MONT KIARA
50480, KUALA LUMPUR
MALAYSIA

PHONE: 006 03 6207 1607
006 03 2782 6991

FAX: 006 03 6207 1697
EMAIL: cscpress@cscjournals.org