SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TNC TRANSACTIONS ON NETWORKS AND COMMUNICATIONS

VOLUME 6, NO. 5
ISSN: 2054 -7420

Load Balanced Network: Design, Implementation and Legal Consideration Issues

Abuonji Paul, Rodrigues Anthony, J., George O. Raburu
School of Informatics and Innovative Systems, Jaramogi Oginga Odinga University of Science and Technology, P. O. Box 210- 40601, Bondo, Kenya.
pabuonji@jooust.ac.ke; tonyr@jooust.ac.ke; graburu@jooust.ac.ke

ABSTRACT

Computer networks have become extremely useful in the modern fast paced work and business environments. Every user of a computing device- desktops, laptops, tablets and mobile phones- wants to connect to a network and communicate with others in near real-time. So networks have become largely ubiquitous. However many challenges exist for perfect network ubiquity. To many users, a network is not useful if they cannot access it when they need it. Many factors such as congestion, disconnection, misconfiguration, network loops, host-source outage and device errors lead to unavailability of a network or network enabled services. This study focused on the design, implementation and legal issues of load balanced networks in order to increase the availability of bandwidth supply. Several network designs were developed and tested situ in a real organization and resulting data used to make decisions on what adjustments to make in the subsequent designs until the best design was achieved in an iterative manner.

Index Terms: load balancing, network, bandwidth, firewall, router.

1 Introduction

Computer networks are very critical in the modern technology driven business environments where they are the facets that tie the entire business processes and applications supporting them into one synchronized and coherent whole. Unlike in the past when computers and networks were mainly used by university researchers for sending and receiving e-mails and by corporate employees for sharing printers [1], the situation has since changed drastically, and currently millions of ordinary citizens use computers and networks for their day to day activities like shopping, banking, studying, communication, entertainment and many others. With the advent and rapid development of social media and mobile digital communication devices, the Internet has become the perpetual global meeting place where young people discover, rediscover and develop their own personalities, influenced by millions of fellow young or older people all over the world [2]. This trend has put much pressure on computer networks and raised the bar significantly high for the caliber of network that would satisfy user needs [3].

Network security now receives unprecedented attention unlike the past. From the perspective of information system security, the three core principles of IS security involve maintaining confidentiality, integrity, and availability of information resources [4]. These three concepts form what is normally referred to as the CIA triad, depicted in figure 1 below. From the standpoint of balanced security [5], some

DOI: 10.14738/tnc.65.5099
Publication Date: 14th September 2018
URL: http://dx.doi.org/10.14738/tnc.65.5099

systems such as those storing trade secrets have critical confidentiality requirements while some like financial transaction values have critical integrity requirements whereas others like e-commerce servers have critical availability requirement. This explains why some organizations opt to change the CIA triad to the AIC triad to underscore the fact that they put more emphasis on availability.

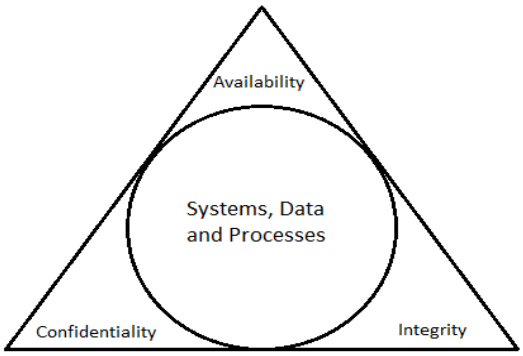


Figure 1: The Information System Security Triad

This viewpoint is further reinforced by the argument that a system must first be available for its confidentiality or integrity to be realized. That is why this diagram places it at the apex of the triangle, and for the same reason, the paper concentrates on how to enhance network security to improve its availability.

2 Related Works

Availability is the assurance that systems and data will reliably be accessed and used whenever needed by authorized users [6]. There are several threats that target system or network availability. They include denial-of-service or distributed denial-of-service attacks, worms, viruses which can clog the whole memory or CPU and render it ineffective and theft of physical computing devices [7]. Security and network administrators therefore need to implement systems with high level of availability. When considering the performance of communication lines, there are four main parameters that one needs to look at namely: bandwidth, delay, jitter and packet loss [8] since in a network, there are applications that require high bandwidth while others are more sensitive to delays or jitters. To succeed in enhancing system availability, an organization must develop appropriate technical mechanisms, security policies and well thought out contracts with external players such as suppliers, contactors and users.

Several controls can be implemented to safeguard availability of an information system and its resources such as redundant array of inexpensive disks (RAID), clustering, load balancing [9], redundant data and power lines, software and data backups, disk shadowing, co-location and off-site facilities, roll-back functions, fail-over configurations and service level agreements [5]. When dealing with outsourced services, the first aspect to consider is the quality of service level agreements (SLA) the organization negotiates and signs with its service providers [10]. Availability is normally computed in terms of mean time to failure (MTTF) and mean time to repair or restore (MTTR) service [11].

$$Node\ Availability = \frac{mttf}{mttf + mttr} \quad (1)$$

Calculating the MTTF and MTTR are fairly simple when dealing with one service like internet bandwidth; however it becomes a little complex when handling sophisticated distributed systems running multiple services. In such a case, to compute availability, those nodes need to regularly log its timestamps, enabling

them to compute their uptimes and downtimes. To compute MTTF and MTTR the node reads the logged timestamps and uses that to calculate the averages for the time between the downtimes. Figure 2 below illustrates the time line of a node experiencing failure, indicating time to failure and time to repair.

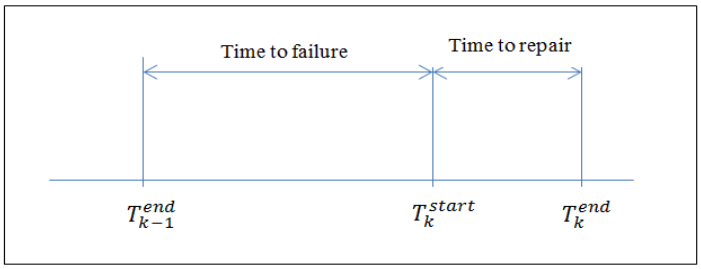


Figure 2: Time line of a node experiencing failure (Tanenbaum & Steen, 2014)

This gives node availability as expressed in the following formula:

$$Node\ Availability = \frac{\sum_{k=1}^{n}(T_k^{start} - T_{k-1}^{end})}{\sum_{k=1}^{n}(T_k^{start} - T_{k-1}^{end}) + \sum_{k=1}^{n}(T_k^{end} - T_{k-1}^{start})} \quad (2)$$

This must meet the minimum required service availability level, according to the SLA. Most high availability SLAs require at least 99.9 %, and can only compromise 0.1 %. Any service outage beyond this may require the service provider to give credit note to the client or pay for damages that occurred during the outage. For accurate data on service availability and down times to be accurately collected, there is need for constant monitoring and logging of the system [11].

The second approach for enhancing availability is by designing and robustly implementing a fault tolerant system. These systems have redundant processors, links, peripherals and software with fail-over or load balancing capacities [12]. Such systems can provide fail-safe capabilities that can enable them to operate at reasonable levels even if there is a major software or hardware failure. Redundancy and replication allows the system to have multiple components that can perform the same task so that if one component fails, the other components can take over the services. However, the limitation is that some components may remain idle while other are being used, thereby bringing about superfluous costs. An example is an organization that engages two internet service providers at the same time in order to mitigate the effects of service outages. If they opt for “redundant” or “backup link” approach, one link will be used until such a time when it fails then the second line will be plugged in. This approach has the advantage of being easy to implement, but very costly due to underutilization of the procured resources.

Another approach is the load balanced mode. Singh and Gangwar [13] defined load balancing as a methodology for distributing workload across multiple computers or other resources over the network links. When applied to management of network traffic, this approach allows multiple routes to the same place to be assigned to the traffic and will cause traffic to be distributed appropriately over those routes [14]. Suppose an organization has subscribed to bandwidth of 20 mbps form two different ISPs- where each ISP provides half of the total bandwidth, the system will be configured such that both the two ISPs supply their bandwidths to a common intelligent device like router or firewall with load balancing capabilities. The bandwidth will be distributed to all users from the common pool while the users will not know which of the two links is transmitting their data. However, when one link fails, all the users will automatically and seamlessly be redirected by the router or firewall to the existing link. All this should be

transparent to the users except that they may notice a drop in transmission speed due to the fact that the available bandwidth is half the original bandwidth. This is illustrated in figure 3 below.

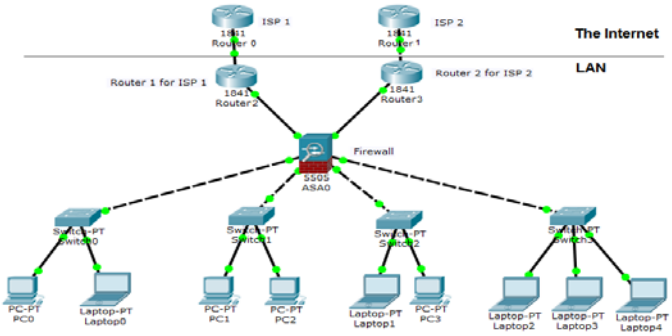


Figure 3: An Illustration of Bandwidth Supply in Load Balanced Mode

The system will need to be configured with two gateway IP addresses and domain IP addresses from both the two ISPs. Supposing the gateway IP for ISP 1 is 64.25.201.140 and that of ISP 2 is 41.206.114. 23, then the pseudo code for the gateway failover will be as follows:

If…
Not able to ping on IP Address "64.25.201.140"
Then…
Shift to another Available gateway

In this case the alternative gateway is 41.206.114. 23. Here none of the gateways should be set as backup because they are expected to be active all the time, and should be given the same weight if the bandwidth procured from each ISP is the same. The failover timeout must also be configured at reasonable time – that is neither too short nor too long for maximum efficiency. Too short a time will create high overhead due to frequent switching between gateway and a longer time may also create some window of service outage when one link fails but the other has not picked up.

The desired failover time can be expressed as follows:

$$Optimum\ failover\ time = \ t_{fmin} < t_f < t_{fmax} \tag{3}$$

Where:

t_{fmin} is the minimum time set for failover.
t_{fmax} is the maximum time set for failover.
t_f is the optimal time which must neither be too short nor too long.

Note that this design is meant to automate link stability vigilance and can greatly improve performance since it is self-organizing or self-adjusting [15]. The system should also log the events including up-times and down-times thereby making it easy to monitor and enforce the SLA. Much as this system looks better and more effective in improving availability, it requires more sophisticated network tools and advanced technical skills on the part of security or network administrators.

Many algorithms exist for load balancing implementation. They are normally referred to as packet scheduling algorithms. Patel and Dalal [16] broadly classifies them into time stamp based and round robin based algorithms. They explain that the latter is more efficient than the former because it eliminates time

stamping and sorting overheads. Vashistha and Jayswal [17] broadly classified load balancing algorithms as static and dynamic. Under static algorithms, they listed round robin, randomized, central manager and threshold algorithms. While examples of dynamic algorithms included sender initiative, receiver initiative, symmetrically and periodically exchanged, central queue and local queue algorithms. Whereas Elngomi and Khanfar [18] also classified the algorithms the same way, they gave central queue and local queue algorithms as the only examples of dynamic algorithms.

However Kaur and Kaur [19] took a different approach. They outlined many load balancing algorithms without any classification. Their list included round robin, weighted round robin, throttled load balancer, active monitoring load balancer, adaptive resource allocation and skewness algorithms. Ray and Sarkar [20] also took a similar approach and listed load balancing algorithms as token ring, round robin, weighted round robin, randomized, central queue algorithm and connection based mechanism.

Several studies have been conducted to assess the performance of various load balancing algorithms. For example, Singh and Gangwar [13] compared the performance of round robin, active monitoring and throttled load balancer in a virtual environment in terms of response time. Their experimental results showed that if they increased the number of datacenter computers, this led to increase in overall average response time in all the algorithms under study. However, the increase in response time was least in throttled load balancing algorithm. This showed that it had better performance. However the algorithms were not tested for their effectiveness in managing network traffic.

Another comparative study was done by Sharma, Singh, and Sharma [21], and a similar one by Vashistha and Jayswal [22]. They identified performance parameters against which to test the algorithms. The parameters included: overload rejection, fault tolerance, forecasting accuracy, stability, centralized or decentralized decision making, nature of load balancing algorithms, cooperativeness of processors or hosts, process migration and resource utilization. The performance comparison is a shown in table 1 below. In summary, static load balancing algorithms were more stable, more accurate at forecasting and use fewer resources.

Table 1: Comparative Study of Performance of Various Load Balancing Algorithms: (Vashistha & Jayswal, 2013)

Parameters	Round Robin	Random	Local Queue	Central Queue	Central Manager	Threshold
Overload Rejection	No	No	Yes	Yes	No	No
Fault Tolerance	No	No	Yes	Yes	Yes	No
Forecasting Accuracy	More	More	Less	Less	More	More
Stability	Large	Large	Small	Small	Large	Large
Centralized/ Decentralized	D	D	D	C	C	D
Dynamic/ Static	S	S	Dy	Dy	S	S
Cooperative	No	No	Yes	Yes	Yes	Yes
Process Migration	No	No	Yes	No	No	No
Resource Utilization	Less	Less	More	Less	Less	Less

A review conducted by Elngomi and Khanfar [23] concluded that there is no particular load balancing technique that fits all information systems. Therefore system designers and developers must be able to carefully choose the technique that is suitable to a given system architecture and it requirements. For

instance if the uncompromisable requirement of the system is stability, forecasting accuracy and optimum utilization of resources, then we must choose a static load balancing algorithm regardless of other performance indicators.

It is for this reason that most modern UTM systems with inbuilt load balancing capabilities use weighted round robin algorithm (WRR). This is a Round Robin based scheduling algorithm used in packet-switched networks with static weight assigned to queues of various connections. It was designed to mitigate the failures of the original round robin algorithm [24]. In this algorithm each connection is assigned a weight and those with higher weight receive more traffic than those with lower weights. But in a situation where all the weights are equal, all connections will receive uniformly balanced traffic. Figure 4 below illustrates how WRR works.

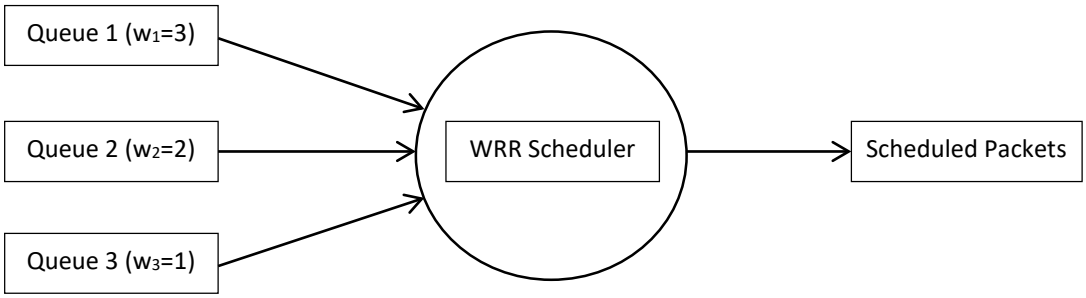


Figure 4: Weighted Round Robin (WRR) Algorithm

The algorithm allocates time for each queue and allows each to transmit data packets based on its weight. By so doing, it ensure that high priority queues do not suffer equal completion from lower priority queues but at the same time prevents lower priority queues from being starved of bandwidth for a long time. It has processing or computation complexity of O(1), thus making it feasible for high speed interfaces in both core and at the edge of computer networks [16].

WRR scheduling is based on assigning a fraction weight $\emptyset_i$ to each service queue such that the sum of weights of all service queues is equal to one.

$$\sum_{i=1}^{N} \emptyset_i = 1 \tag{4}$$

Considering that the weight is a fraction of the total number of packets to the transmitted, and we need to determine the number of integer packets to be transmitted from each queue, the fraction weight is then multiplied by a constant integer M. and the product is rounded off to nearest larger integer to obtain integer weight w_i. This integer weight value of each queue specifies number of packets to be transmitted from that queue. The total sum of these counter values is referred to as round robin length. Therefore the integer weight of i_{th} queue is:

$$w_i = \lceil \emptyset_i * M \rceil \tag{5}$$

The sum of existing *N* active connections in the network is defined as round robin length *W* and is given by the following formula

$$W = \sum_{i=1}^{N} w_i = M \tag{6}$$

As earlier stated, this algorithm provides stability, forecasting accuracy and good resource utilization. It has a drawback which is common to most scheduling algorithms that latency is affected by transmission rate of output link and the number of connections [16].

3 Methodology

The study adopted descriptive and diagnostic research design. Descriptive study was used to collect and record primary data depicting the problems, issues or concerns within the system under study [25], [26] while diagnostic study was used to facilitate an in-depth analysis of the research variables by first investigating the root cause of the problem and dealing with it in terms of emergence of the problem, diagnosis of the problem, solution for the problem and where no concrete solution has been found, a suggestion for the problem solution or escalation [27]. A network infrastructure was designed and deployed in a university. The process was guided by the Prepare Plan Design Implement Operate Optimize (PPDIOO) Network Life Cycle [28]. This design methodology was adopted because it was developed to support evolving networks like the one used in the research. Three Cisco routers, several switches and a cyberoam firewall were configured and used at various levels of the study. Two ISPs supplied internet bandwidth. The supply and use of the bandwidth was monitored to test availability. In-situ data was captured by real-time bandwidth monitoring tools deployed at the gateway to monitor bandwidth availability in terms of stability and amount supplied. This data was presented graphically and compared for different designs and results used to make decisions on what adjustments were supposed to be made on the current design.

4 Load Balanced Network Topologies and Architectures

Several network topologies and architectures were developed and tested for security and bandwidth load balancing using two ISPs.

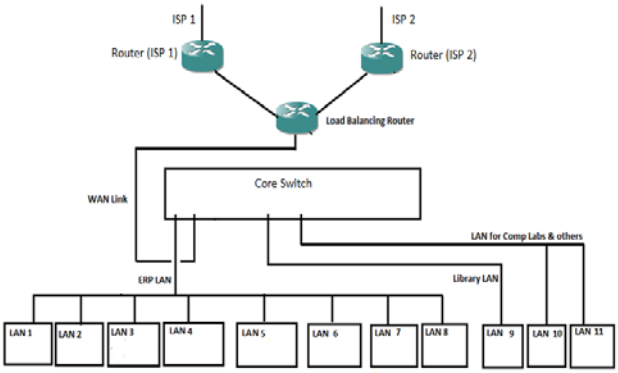


Figure 5 (a): LAN Topology with two ISPs and Load Balancing on Router

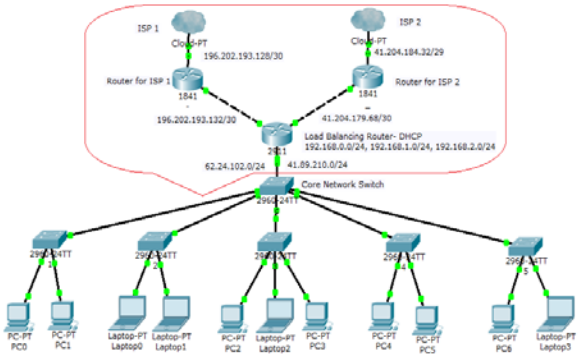


Figure 5 (b): LAN Architecture with two ISPs and Load Balancing on Router

Figures 5 (a) and (b) above illustrate a topology and architecture respectively of a network with two ISPs, three routers and a core network PnP switch. In this arrangement, the link from each ISP terminates on a different router. Then the third router is dedicated to perform load balancing functions such as aggregation of traffic from or to the two ISPs. The load balancing router performs DHCP, NATing and routing functions. This arrangement though was effective in improving link availability still left the LAN in desperate need for security- both at the gateway and within.

As illustrated in figures 6 and 7, this configuration worked well except that the network was vulnerable to cybercrime activities since there was no firewall. The load balancing router performed NATing, DHCP as well as managing the traffic load on both links. This setup was a major achievement as it improved availability by providing two simultaneous links- whereby if one link fails, all the users' traffic is seamlessly and transparently routed in and out through the remaining link. However the main concern was that the absence of firewall still exposed the network to numerous external threats. Besides, the network still had one broadcast domain containing three pools of /24 IP addresses provided by three DHCP servers with a single default gateway- 192.168.2.0. There was dire need to address these issues so a new design was proposed.

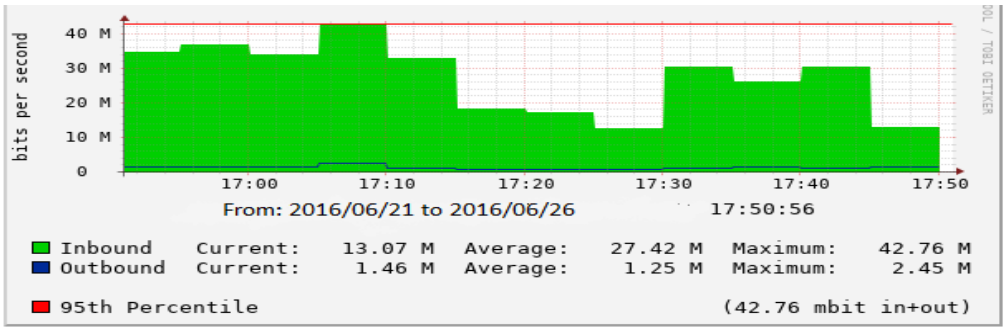


Figure 6: Traffic on ISP 1

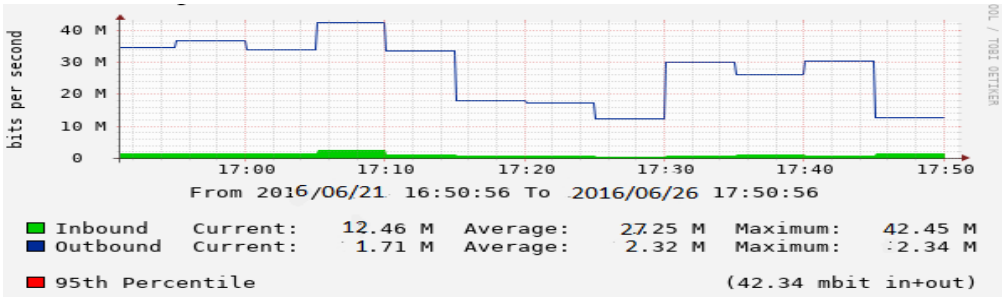


Figure 7: Traffic on ISP 2

The need for security triggered a desire to develop a system that would provide a high availability link as well as border and internal LAN security. This led to the topology and architecture in figures 8 (a) and (b) below respectively. In this set up, the load balancing was done by the third router that aggregates traffic from/ to both ISPs. However, instead of this router feeding into the LAN directly, it was connected to a UTM configured in gateway mode.

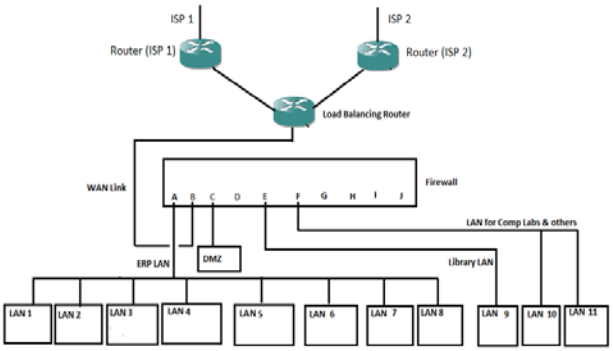


Figure 8 (a): LAN Topology with two ISPs, three routers and Firewall

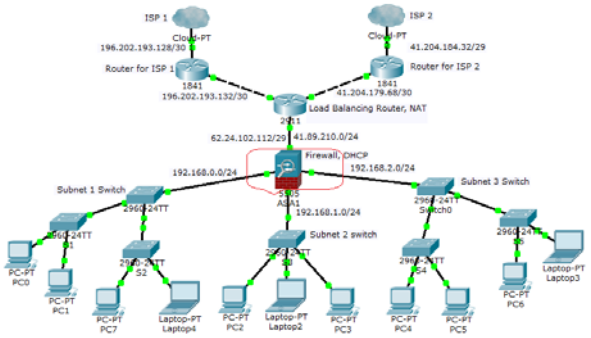
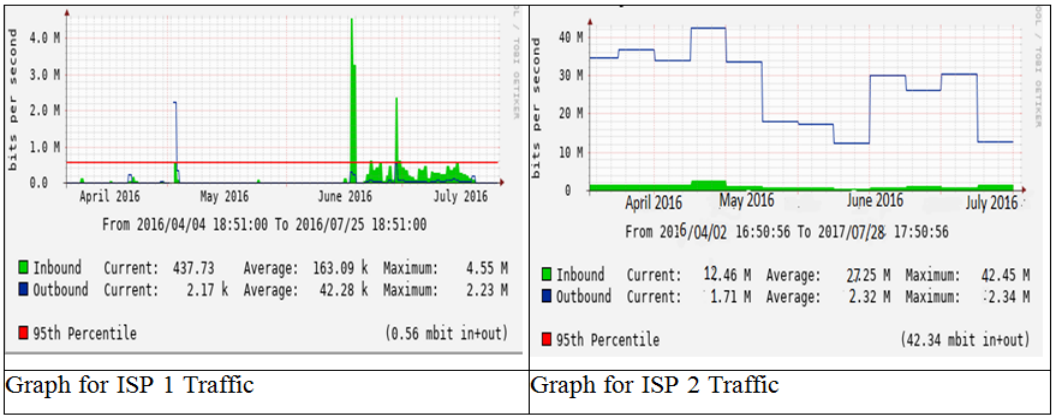


Figure 8 (b): LAN Architecture with two ISPs, three routers and Firewall

Even though this design improved the security of the network, it had considerable operational challenges due to configurational intricacies. The UTM was implemented in gateway mode in order to utilize all its required features (as opposed to bridge mode which is limited in many ways). In this gateway mode, the UTM could take only one IP address as gateway on its WAN interface. This meant that either (not both) of the ISP gateways could be used at a time. The UTM was configured to provide security as well as perform NATing, subnetting and DHCP functions. However load balancing activities were supposed to be performed by the "load balancing router" as was the case in design 2 (a). The justification of this design was to distribute network tasks so that no single device is overworked since this could reduce network efficiency, after all the intention of the design was distribution and balancing of network load. Nevertheless, this design did not work as expected. Figure 9 below shows graphs illustrating how traffic was transmitted through the two network links ISP 1 and ISP 2 if the gateway on UTM is from ISP 2.



Graph for ISP 1 Traffic	Graph for ISP 2 Traffic

Figure 9: Data Traffic in Design flow

It was observed that if the gateway on the WAN interface of the UTM was from ISP 2, all the date exiting the LAN would be transmitted through ISP 2's network and vice versa. Even though the load balancing configuration on the router remained the same, the added layer of security and associated configurations had inadvertently invalidated some aspects of the routers configurations. With this configuration intact, the only way to remedy the situation was to configure other upstream routers of the two ISPs to allow Border Gateway Protocol (BGP) to enable the exchange of routing and reachability information between the two autonomous ISP systems involved. However this would have been a complex process which required even more sophisticated configurations, prolonged boardroom negotiations and possible legal

implications for one of the two ISPs since it was not strictly a commercial ISP but one that supplied bandwidth to educational institutions and recognized as such by the government.

This challenge motivated the researcher to design another topology and architecture that would take advantage of the salient feature of the previous set ups but mitigate their weaknesses. Figures 10 (a) and (b) below depicts the new arrangement. A public switch was introduced between the load balancing router and the UTM. Date emanating from the two ISPs was aggregated at the load balancing router and then passed on to the public switch which was connected to two different WAN interfaces of the UTM.

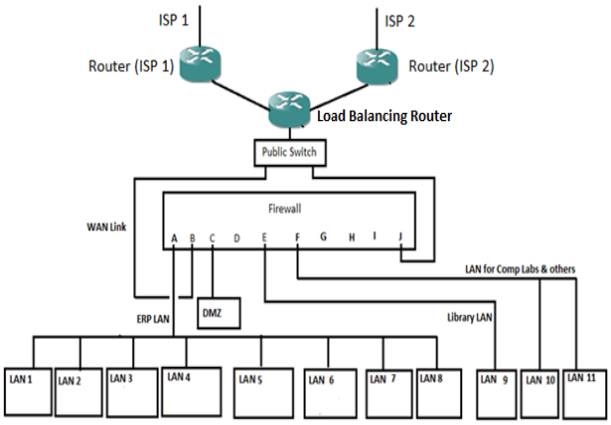


Figure 10 (a): LAN Topology with two ISPs, three routers, firewall and public switch

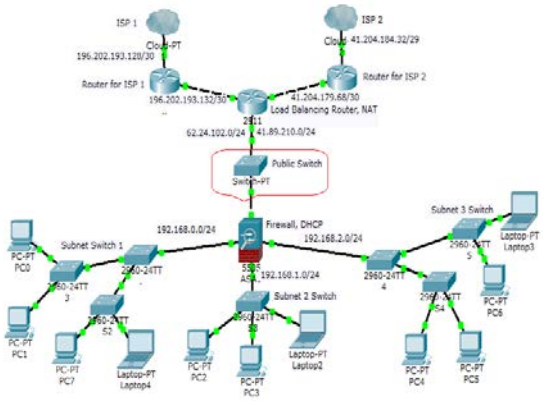


Figure 10 (b): LAN architecture with two ISPs, three routers, firewall and public switch

This topology initially alleviated some of the problems of the previous topologies. At first the internet access speed from the LAN increased tremendously as was expected since the two links were being used simultaneously and the LAN had also been subnetted thereby creating many smaller collision domains and decreasing the broadcast domain. However this was as far as the success went. This setup added more confusion to the system. Byzantine and transient faults characterized the design. For example in one office with six users, four users would get internet access whilst two could not. When one removed the Ethernet cable from the NIC port for a short while and reconnected it then that would solve the problem in one case but did not in another case. Technically speaking, byzantine faults are difficult to troubleshoot and solve since it is difficult to identify the lucid cause. Because of these problems, a new design was developed and implemented.

5 The Ideal Load Balanced Network for Security and High Availability

Due to the problems described above, a fundamentally different design was developed to solve them. This structure used two routers and the UTM. The load balancing function was configured in the UTM instead of the router. Other functions such as NATing, DHCP, subnetting and security were also configured on the UTM. The downside was that this configuration overloaded one devise. However the upside is that this was the configuration that could effectively solve the problem. The topology and architecture are illustrated in figures 11 (a) and (b) below respectively.

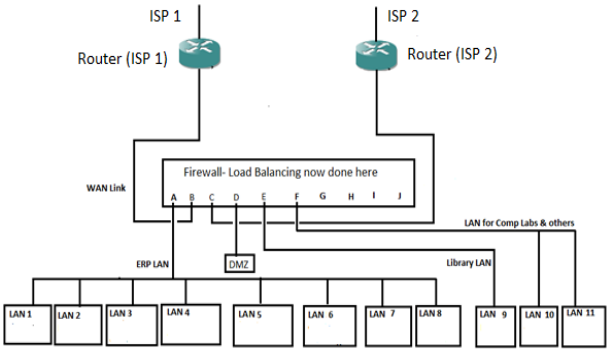


Figure 5 (a): LAN topology with two ISPs, two routers and firewall

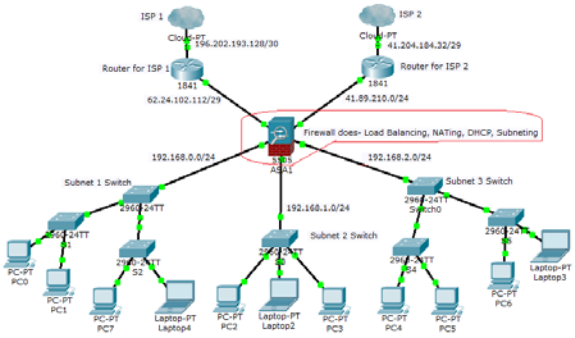
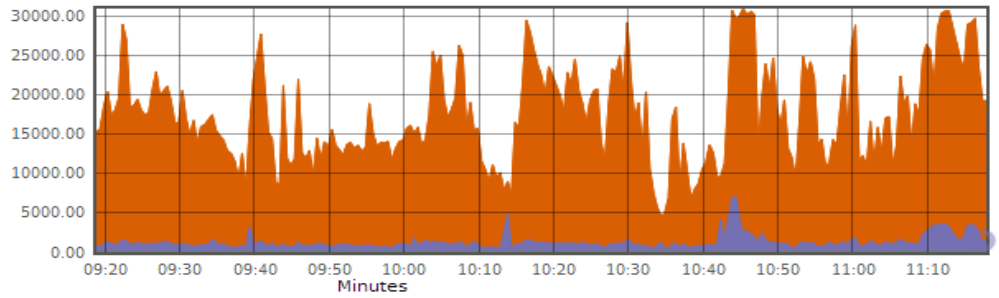


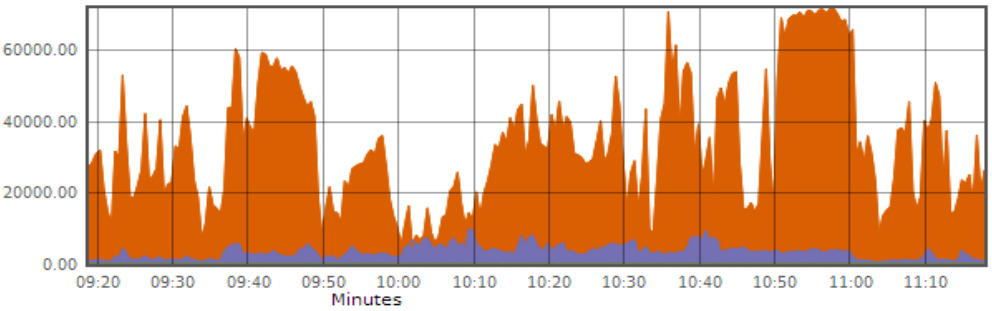
Figure 5 (b): LAN architecture with two ISPs, two routers and firewall

Figures 12 and 13 below are graphs viewed from the UTM WAN interfaces illustrating the transmission of traffic through the two network links ISP 1 and ISP 2 respectively.



	Max	Min	Average	Current
Received KBits/Sec	31014.59	4607.17	17606.12	19248.01
Transmitted KBits/Sec	7113.19	483.94	1306.33	1462.53

Figure 12: Data Transmission through ISP 1



	Max	Min	Average	Current
Received KBits/Sec	71987.23	4847.66	33916.35	26455.15
Transmitted KBits/Sec	9821.67	536.77	3389.52	1053.05

Figure 13: Data Transmission through ISP 2

The graphs demonstrated that the load was evenly distributed between the two internet links as was expected. This design provided both security and load balancing services to the LAN. However it still had a few limitations which were ingrained in design weakness of the UTM used. One major weakness was that whereas the UTM provided for many simultaneous WAN connections, its design only had provision for three DNS IP addresses as opposed to four required for a good UTM that can take two ISP connections. The second weakness emanated from the algorithm used, as illustrated in the following pseudo code.

If…
Not able to ping on IP Address "64.25.201.140"
Then…
Shift to another Available gateway

Note that there are instances when the internet bandwidth may not be available but the ISP's gateway is reachable. In such cases this pseudo code states that the UTM users connected through that particular gateway should not be switched over to another gateway. This explains why there were instances in the network when one connection failed and the failover to the next gateway was not seamless. Figure 14 below illustrate three different scenarios.

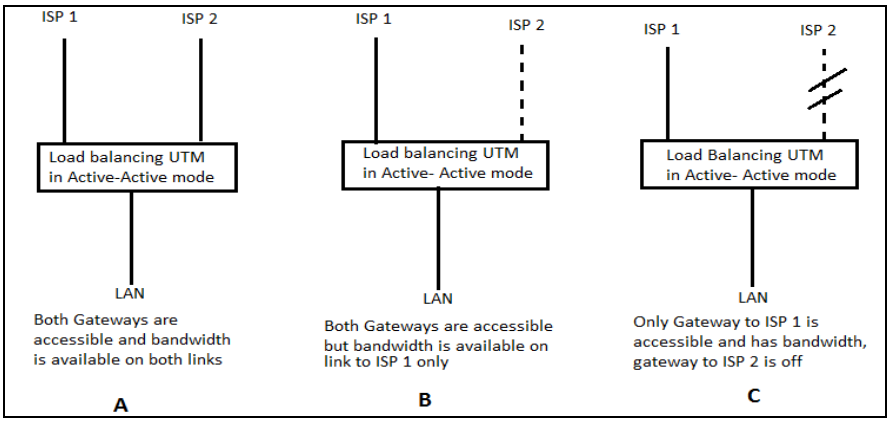


Figure 14: Load Balancing Failover Scenarios

In scenario A, internet users in the LAN are all expected access the internet as they required. The load will be distributed to the two ISP links by the UTM in the ratio of 1:1. In scenario B, the internet users who were connected to ISP 1 will access the internet as they required, but the users connected to ISP 2 will not access the internet until either of the following two actions are taken: (i) they disconnect their Ethernet cables from their computers NICs for a short while and reconnect them so that the UTM connects them to the internet through ISP 1 or (ii) the network administrator disconnects the cable connecting the UTM to ISP 2 so that the UTM will not be able to ping the gateway of ISP 2, in which case all the internet users in the LAN will be connected to ISP 1. Finally in scenario C, as soon as the gateway to ISP 2 goes off, the UTM will seamlessly and automatically connect all internet users in the LAN to ISP 1.

6 Discussion

There are many factors that affect the stability of bandwidth supply in a network. Some of the factors are beyond the control of the organization receiving the service and thus the guaranteed can only be got by negotiating water-tight contracts and SLAs with the service providers. However for those organizations that operate in remote places, far away from the internet backbone cable, it is prudent to adopt link replication as a reinforcement to the administrative controls to enhance link availability. In this process, link redundancy acquired for backup in not advisable due to superfluous costs that come with it- for example the primary link will be used exclusively or largely while the backup or secondary link is either left idle or underutilized until the primary link fails. The ideal approach therefore is load balanced connection which, even though comes with advanced and sophisticated technological demands, will provide a better solution at economically justifiable cost. Each of the four designs described above has its strengths and weaknesses as earlier explained. The final design depicts a network in which bandwidth

from the two links are aggregated and redistributed in a LAN network that is also protected by a gateway firewall.

7 Conclusion

We have demonstrated that if an organization requires high internet bandwidth availability in its LAN which is protected by a gateway or border firewall and therefore decides to be connected to two IPSs simultaneously whereby one of the ISPs is a commercial ISP (ISP 1) but the other is not an ordinary commercial ISP (ISP 2) which is restricted from carrying any traffic of commercial ISPs by its contractual or legal obligation to a third party such as the government or any other regulatory authority, so that ISP 2 cannot configure its BGP policies to allow traffic from ISP 1 to transit through ISP 2's network, then the best configuration for load balancing is to use a firewall with load balancing capabilities.

REFERENCES

[1] Tanenbaum, A. S. (2011). Computer Networks; 4th ed. Prentice-Hall, Inc: New Jersey.

[2] Greenwald, G. (2014), No Place to Hide: Edward Snowden, the NSA & the Surveillance State; Penguin Random House, UK.

[3] Membrey, P., Plugge, E. & Hows, D. (2012); Practical Load Balancing: Ride the Performance Tiger: The ACM Digital Library; Apress Berkely, CA, USA ©2012 ISBN:1430236809 9781430236801

[4] Cocca, P. (2004). SANS Institute InfoSec Reading Room: Email Security Threats. Retrieved on 17th November, 2012 from: http://www.sans.org/reading_room/whitepapers/email/emailsecurity_threats_1540

[5] Harris, S. (2013), All in One CISSP. McGrow-Hill: New York

[6] Stallings, W. (2011). Network Security Essentials: Applications and Standards, 4th Ed; Pearson Education, Inc: Prentice Hall

[7] Laudon, K. C. & Laudon, J. P. (2012). Management Information Systems: Managing the Digital Firm, 12th ed. Pearson Education Limited: Edinburgh Gate, Harlow.

[8] Orzach, Y. (2013), Network Analysis using Wireshark Cookbook; PackT Publishing: Birmingham- Mumbai.

[9] Zhang, J. *et al*. (2018), Load Balancing in Data Center Networks: A Survey; *IEEE Communications Surveys & Tutorials (Early Access); Electronic ISSN: 1553-877X; CD-ROM ISSN: 2373-745X. Retrieved on: 15th August, 2018; from: https://ieeexplore.ieee.org/document/8316818/*

[10] Stewart, J. M., Tittel, E. & Chapple, M. (2005), CISSP: Certified Information Systems Security Professional Study Guide; 3rd ed. Sybex Inc.: London

[11] Tanenbaum, A. S. & Steen, M. V. (2014), Distributed Systems: Principles and Paradigms, 2nd ed. Edinburg Gate: Pearson Education Limited.

[12] O'Brien, J. A. & Marakas, G. M. (2011). Management Information Systems, 10th ed. McGrow- Hill/ Irwin: New York.

[13] Singh, H. & Gangwar, R. C. (2014), Comparative Study of Load Balancing Algorithms in Cloud Environment: *International Journal on Recent and Innovation Trends in Computing and Communication; ISSN: 2321-8169; Volume: 2 Issue: 10; PP: 3195 – 3199*

[14] Peterson, L. L. & Davie, B. S. (2007). Computer Networks: A systems Approach, 4th ed. Elsevier, Inc.: San Francisco.

[15] Ma, Y., Chen, J. & Lin, C. (2018); Automated Network Load Balancing and Capacity Enhancing Mechanism in Future Network: *IEEE Access, Volume 6.*

[16] Patel, Z. & Dalal, U. (2014), Design and Implementation of Low Latency Weighted Round Robin (LLWRR) Scheduling for High Speed Networks: *International Journal of Wireless &Mobile Networks (IJWMN):* Vol. 6, No. 4.

[17] Vashistha, J. & Jayswal, A. K. (2013), Comparative Study of Load Balancing Algorithms: *IOSR Journal of Engineering (IOSRJEN) e-ISSN: 2250-3021, p-ISSN: 2278-8719 Vol. 3, Issue 3; PP 45-50*

[18] Elngomi, Z. M. & Khanfar, K. (2016), A Comparative Study of Load Balancing Algorithms: A Review Paper: International Journal of Computer Science and Mobile Computing; *Vol. 5, Issue. 6, June 2016, pg.448 – 458*

[19] Kaur, P. & Kaur, D. (2015), Efficient and Enhanced Load Balancing Algorithms in Cloud Computing: *International Journal of Grid Distribution Computing:* Vol.8, No.2 (2015), pp. 9-14.

[20] Ray, S. & Sarkar, A. D. (2012), Execution Analysis of Load Balancing Algorithms in Cloud Computing Environment: International Journal on Cloud Computing: Services and Architecture (IJCCSA), Vol.2, No.5, October 2012.

[21] Sharma, S., Singh, S. & Sharma, M. (2008), Performance Analysis of Load Balancing Algorithms: *International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol. 2, No: 2, 2008.*

[22] Vashistha, J. & Jayswal, A. K. (2013), Comparative Study of Load Balancing Algorithms: *IOSR Journal of Engineering (IOSRJEN) e-ISSN: 2250-3021, p-ISSN: 2278-8719 Vol. 3, Issue 3; PP 45-50*

[23] Elngomi, Z. M. & Khanfar, K. (2016), A Comparative Study of Load Balancing Algorithms: A Review Paper: International Journal of Computer Science and Mobile Computing; *Vol. 5, Issue. 6, June 2016, pg.448 – 458.*

[24] Daryapurkar, A. & Deshmukh, V. M. (2013), Efficient Load Balancing Algorithm in Cloud Environment: *International Journal Of Computer Science And Applications:* Vol. 6, No.2

[25] Kothari, C. R. & Garg, G. (2014): Research methodology: Methods and techniques. New Delhi: New Age International (P) Ltd, Publishers.

[26] Nzioki, P.M., Kimeli, S. K., Abudho, M. R., Nthiwa, J. M. (2013): Management of working capital and its effects on profitability of manufacturing companies listed at NSE, Kenya: *International Journal of Business and Financial Management Research*, *1:35 - 42.*

[27] Farooq, U. (2013), Types of Research Design. *Referred Academic Journal, 08:21*

[28] Cisco Study Guide (2007), Designing Cisco Network Service Architecture. Retrieved on 24th June 2017, from: www.itsolutions.pro%2Fimages%2Fstories%2Fdocs%2Fdesigningcisconetworkservicearchitecturesarchv2.0volume1.pdf&usg=AFQjCNEjONV tN0daSRWcsOfDYIDc0FOpbw