



Load Balancing the Network Traffic in the Nth Mode of IPtables

ANUPAMA, O.P. GUPTA

[#]School of Electrical Engineering & Information Technology, Punjab Agricultural
University PO Box 141004, Ludhiana, India.

<http://dx.doi.org/10.13005/ojcs/9.02.11>

(Received: May 18, 2016; Accepted: June 20, 2016)

ABSTRACT

In this paper, we present the network load balancing to optimize the network interfaces that are present in the cluster. We know, everything belongs to internet now a days. With the increase in the popularity of the web, network traffic increases and need for fast access to the internet is also on demand. For smooth services, redundancy of the servers is required. As a result, load balancing has become a very critical job. A single server is not able handle all the traffic which results into a server crash. Network is divided into different paths on which the transmission of packets is done. They are distributed on all the interfaces simultaneously. We investigate the performance of per-packet load balancing in different servers. Simulation results reveal that it is always beneficial to use load balancing and backup of other servers are always present in case of any failure in network.

Keywords: Load balancing, IP tables, Equal-cost paths

INTRODUCTION

One of the main issues in the development of network applications is the load balancing. For instance, a social networking site requires so many servers in different areas to fetch the results in order to generate web page. Therefore, it is necessary to balance the load. Load balancing is the ability of the router to distribute the client's request across the number of servers within the cluster to reach the destination address. It is a network service which controls the network servers and network

resource utilization. The load is distributed on the network among N paths which have been established between the routers. The request is distributed on the basis of network variabilities such as IP addresses and destination port.

When the client requests, network load balancing transparently distributes the request to servers but from client side of view it acts as a single sever that is answering to its request. Multiple number of servers and ISPs (Internet Service Providers) are connected together to form a cluster

and it appears to be a single server to the client. User's request is routed to the best path among several paths that are involved in the network to provide the same service. It also enables the host to remove the server from the cluster when the load goes down and it also has the ability to add host when the load goes up without bringing cluster goes down. It provides high availability. The network is always available without any delay, by detecting host failures and automatically redistributing the network traffic to other working servers. Servers can be positioned anywhere in the network and there is no restriction on the number of servers. In case of network failure, it will only reduce the number of servers available in the network. Load balancing improves the performance of the server and there is proper utilization of the network interfaces present in the cluster, which further increases the bandwidth of the network. It enhances the network performance.

LITERATURE REVIEW

Baldine *et al* (1998) studied the dynamic load balancing problem in broadcast WDN networks. The tunable transceivers take a non-negotiable amount of time to switch between wavelengths so as to check which part of network is unavailable for normal operation. Their main contribution is a new approximation algorithm for load balancing problem that provides for trade off selection between two conflicting goals using two parameters¹.

Kashem *et al* (2000) explained that the reconfiguration of the network for load balancing determined the switching options for particular set of loads relieved overloads on distributed system and improves load balancing. The test results showed that the proposed method can be identified as the most effective branch-exchange operations for load balancing with less computational efforts and time. In the proposed method, the number of load flow solutions has been reduced to a greater extent².

Du *et al* (2003) investigated a novel distributed load balancing scheme for cellular networks that changes the cellular coverage in real time according to geographical distribution. In this paper, a cooperative negotiation approach for cellular network coverage with semi-smart antennas

is described. Optimum local coverage agreements can be reached by the use of the negotiations between adjacent base stations³.

Hoang *et al* (2006) expected that future ITS (Intelligent transport system) network will consists of a lot of mobile equipments and a complex network is formed. In this paper, source node performed the load balancing on the probe packets sent by the destination node. So the number of probe packets will reduce to half. They also represented a new formula for available bandwidth estimation of 802.11 ad hoc networks⁴.

Hayian *et al* (2009) proposed multi-ISP load balancing optimization model based on BP neural networks in order to solve the problem of manual strategy choice when they use the campus network of multiple internet service provider (ISP). The establishment of network topology structure was discussed in this paper and a detailed description about the data and processing were given. The effectiveness of the model was verified by the experimental results⁵.

Zhang (2011) presented an algorithm on an exponentially hard problem that is load balancing. To improve the performance of wired-wireless networks they presented this algorithm. This algorithm identifies the pair of access point for load balancing and pair of access points are divided into segments, so that the required search space is reduced. Average throughput and packet delivery ratio were shown after load balancing is performed⁶.

Saxena *et al* (2012) explained that tremendous growth has been witnessed in the present era and different applications are supported by it. There is an enormous pressure on Internet Service Providers (ISPs) to make available adequate services for the traffics like VoIP and video on demand. Since the resources like computing power, bandwidth etc. are limited, the traffic needs to be engineered to properly exploit them⁷.

Wang *et al* (2013) described that due to the current number of domestic Internet access provider (ISP), backbone operators are distributed in different regions which slows down the access

investigated the effects of out of delivery of packets on throughput and flow completion time of long and short flows respectively. The presence of out of order packets in a DCN using per-packet load balancing also confirmed in their simulations. In spite of the presence of out of packet delivery simulations should be such that the per packet load balancing may yield smaller average flow completion time for short¹⁰.

Fig. 2: Network Layout

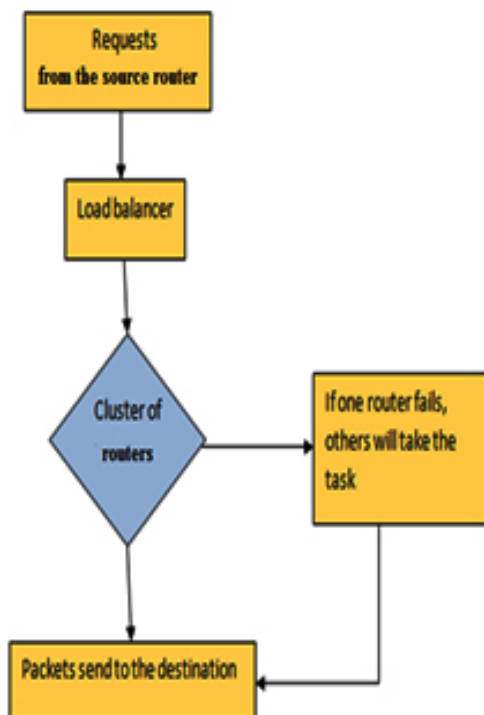


Fig.1: Implementation Flow Chart

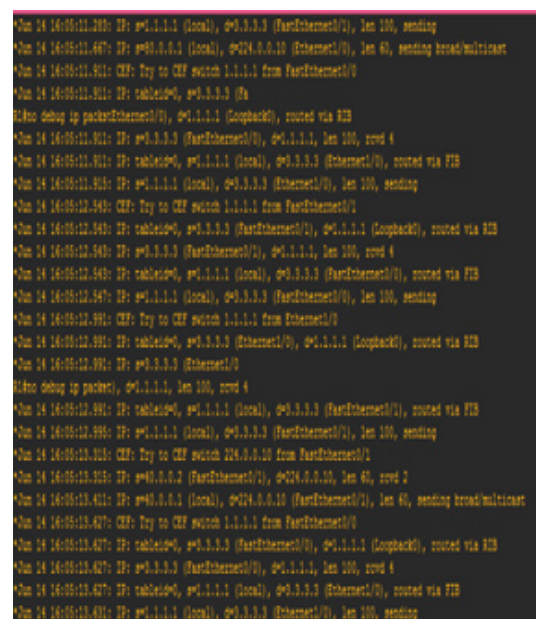


Fig. 3: Debugging shows the load balancing

MATERIAL AND METHODOLOGY

In this section, we describe the methodology for balancing the load. IPtables are the fourth generation Linux packet filtering application. Iptables are used to set up, maintain the tables of the IP packets. It also referred as packet filter. Per-packet load balancing technique is used in it. It works at each node by randomly selecting one of the interfaces to forward each incoming packet. The selection of an interface for an incoming packet is done by using round robin algorithm. Multiple

equal-cost paths to the destination are available. One of them is randomly selected and the packet is transmitted to specific node. This technique also requires the calculation of best path, so that the per-packet load balancing is able to distribute the traffic among them. Fig.1. shows the implementation flow chart.

Fig.2 shows the network layout of 9 routers through which the traffic has to be distributed equally. Node labeled as R1 acts as a client server and node labeled as R4 acts a host server. In

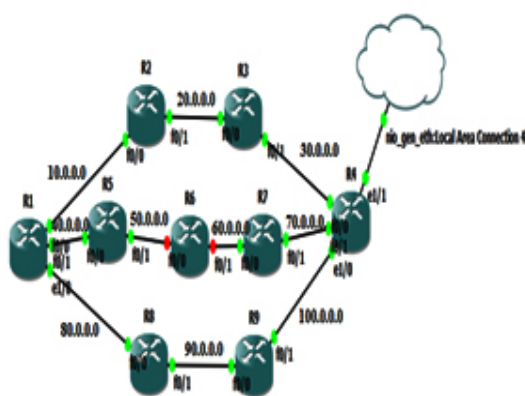


Fig. 4: Network layout during node failure

```

*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (Interface0), v0.1.1.1 (Interface0), route via R3
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (Interface0), v0.1.1.1, len 100, send 1
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (local), v0.1.1.1 (Interface0), route via R3
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (local), v0.1.1.1 (Interface0), len 100, sending
*ip 10.0.0.0/8: IP: try to get source 1.1.1.1 from Interface0
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (Interface0), v0.1.1
*ip 10.0.0.0/8: route via R3
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (Interface0), v0.1.1.1, len 100, send 1
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (local), v0.1.1.1 (Interface0), route via R3
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (local), v0.1.1.1 (Interface0), len 100, sending
*ip 10.0.0.0/8: IP: try to get source 1.1.1.1 from Interface0
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (Interface0), v0.1.1.1 (Interface0), route via R3
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (Interface0), v0.1.1.1, len 100, send 1
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (local), v0.1.1.1 (Interface0), route via R3
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (local), v0.1.1.1 (Interface0), len 100, sending
*ip 10.0.0.0/8: IP: try to get source 1.1.1.1 from Interface0
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (Interface0), v0.1.1.1 (Interface0), route via R3
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (Interface0), v0.1.1.1, len 100, send 1
*ip 10.0.0.0/8: IP: redacted, v0.1.1.1 (Interface0), v0.1.1.1, len 100, send 1

```

Fig. 5: Debugging shows Redundancy

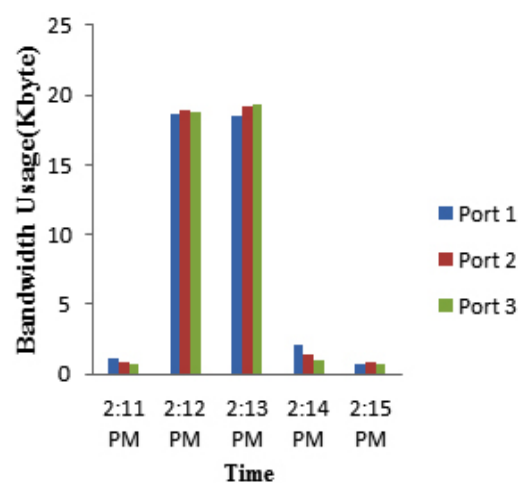


Fig. 6: Comparison bandwidth utilization on all ports for incoming traffic

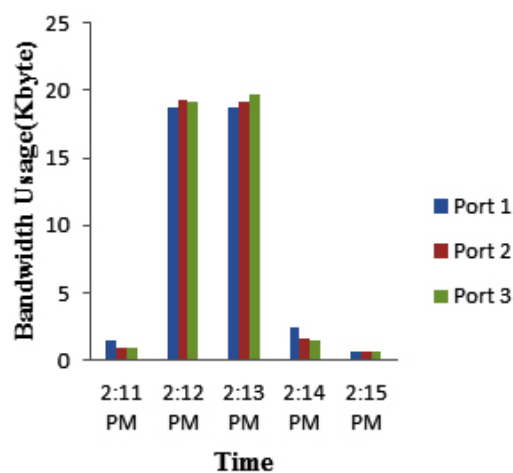


Fig. 7: Comparison of bandwidth utilization on all ports for incoming traffic

between there are multiple paths through which the network traffic is distributed to reach the host machine.

A buffer size of 1000 packets is selected. Every interface has bandwidth of 100Mb/s and delay of 50 μ s, 100 μ s. Point to point links is created and per-packet load balancing at each router is enabled. We also established the best routes by calculating the metrics such as bandwidth and delay.

Now, how this load balancing works. R1node is taken as a client and R4 works as Host server and all other servers acts as workers. All the packets will flow from the loopback address of 1.0.0.0 network that is build on R1 to 3.0.0.0 network of loopback address which is build on R4 and vice versa. First packet is randomly transmitted on one of the best route and then second packet on second route, third one on third route. Then again

Table 1: parameters Used In Simulation

Parameters	Values
Number of nodes	9
Bandwidth	100 Mb/s
Delay at links	50 μ s, 100 μ s
Number of packets	1000

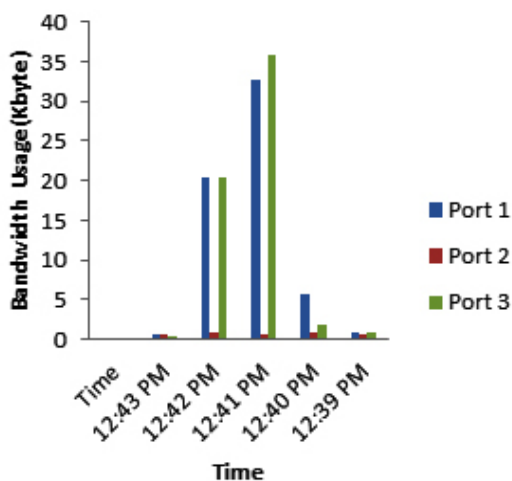


Fig. 8: Comparison of bandwidth utilization on all ports for incoming traffic during redundancy

it send packet on the route that we used first and hence, it follows round robin algorithm.

This figure shows the equal distribution of packets on all the interfaces and the packets are in order as they are distributed from the client to reach the host as its destination.

For smooth services, redundancy is required. Such as if there is a network failure in the first interface, there is always a backup of another

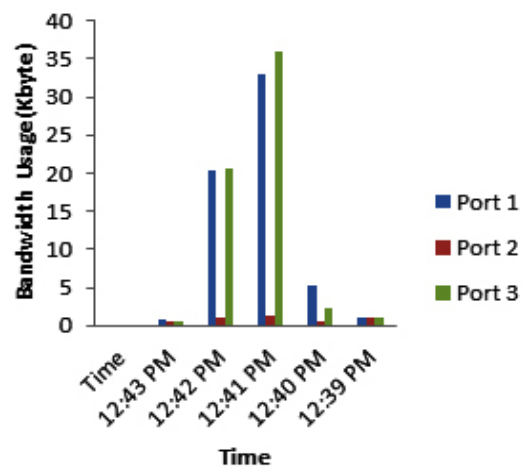


Fig. 9: Comparison of bandwidth utilization on all ports for outgoing traffic during redundancy

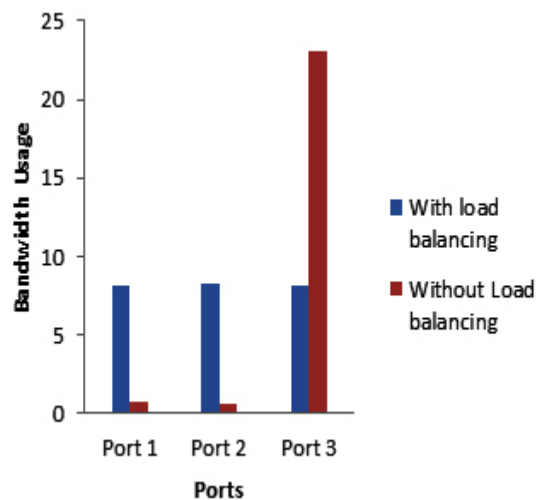


Fig.10: Comparison of average bandwidth usage of traffic IN with load balancing and without load balancing

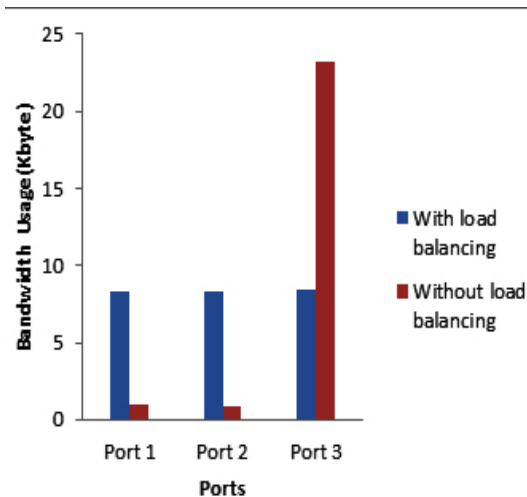


Fig. 11: Comparison of average bandwidth usage of traffic OUT with load balancing and without load balancing

interface to load the traffic. Therefore, it serves the purpose of maintaining the network status always up with multiple paths. Fig.4. shows R6 has stopped working. Red means the routers are stopped and green shows they are working.

Fig 5 shows the debugging of packets. When a server R6 is down, all the traffic goes to other servers through other Fastethernet0/0 and Ethernet 1/0 interface. Client is sending its packets which are transmitted through the servers which are up. The packets are not allowed through the interface Fastethernet0/1. They are automatically transmitted through other nodes without any delay.

Figure 6 & 7 shows the comparison of all the three ports. It shows when is packets are send from client to the server they are equally distributed on all the interfaces for the incoming and outgoing traffic. There is per-packet transmission on all the ports which makes load of the network traffic

balanced. All the nodes are equally utilized and the bandwidth usage of all the nodes is also equal. This shows the load is balanced on all the ports. None of the port is overloaded and none of the port is underloaded. It is improving the performance of all the servers that are balancing the load and there is proper utilization of the network interfaces present in the cluster, which further improving the bandwidth utilization of the network.

Fig.8. shows the comparison of the bandwidth usage on all the ports for incoming traffic while fig.9. shows the comparison of the bandwidth usage on all the ports for the outgoing traffic. These graphs demonstrate that the network traffic going through port 1 and port 3. No traffic is going and coming through port 2 because of the node down on that link. So, there is no loss of data.

Fig.10 and 11 shows the comparison of the average bandwidth usage of incoming and outgoing traffic of the two scenarios of the simulation. In case of load balancing, the average bandwidth usage is more effectively utilized on comparing it with the average bandwidth usage when there is no load balancing. The network traffic will be reduced by distributing it on all the links.

CONCLUSION

It is concluded that when the load is balanced, it improves the network performance which results into the effective utilization of the bandwidth. Performance of the network is degraded and utilization of bandwidth on some links is low in case of no load balancing. Simulation results show that it is always beneficial to use load balancing. It also shows the recovery of the network traffic transmission in case of failure of a server on other paths and it again achieves to load balance on other links. This study can further be conducted by using different parameters and by varying the different parametric values.

REFERENCES

1. Baldine I and Rouskas G N *Dynamic Load Balancing In Broadcast WDM Networks with Tuning Latencies*. Pp 78-85. North Carolina State University, Raleigh (1998).
2. Kashem M A, Ganapathy V and Jamson G B *A Geometric Approach For Three Phase*

- Load Balancing in Distribution Networks*. Pp 293-98. Queensland University Of Technology, Australia (2000).
3. Du L, Bigham J and Cuthbert L *Intelligent Cellular Network Load Balancing Using A Cooperative Negotiation Approach*. Pp 1675-79. University of London, London (2003).
 4. Hoang V D, Shao Z and Fujise M Efficient Load Balancing in MANETs to Improve Network Performance. *Proc 6th ITS Telecom*. Pp 753-56. National Institute of Information Communication technology, Japan (2006).
 5. Haiyan X, Wang J, Depeng Z and Hui S Campus Network Multi-ISP Load Balancing Optimization Model Based on BP Neural Networks. *Proc 2009 Commun Faculty*. pp 611- 15. Dalian Maritime University, Dalian, China (2009).
 6. Saxena K , Kumar R and Chaudhary N S Load Balancing in IP/MPLS Networks: A Survey, © SciRes (2012).
 7. Wang L Multi-outlet network load balancing solution. *Proc 2nd Sci and Social Res*. pp 359- 62. Shanghai Urban Management College Shanghai, China (2013).
 8. Zhang J and Lee Y *A new Load-Balancing Algorithm For Multihop Wireless Networks*. Pp 162-166. Jiangsu University of Science and Technology Zhenjiang, China (2011).
 9. Kaymak Y and Rojas-Cessa R Per-Packet Load Balancing in Data Center Networks. *Proc 36th Sarnoff symposium*. Pp 140-44 (2015).
 10. Bautista J, Gabarron T and Doudane G Y Distributed Load Balancing by Two-Hop Relaying in LTE-Advanced Networks. *Proc 11th Int Conf Sensing, Communic & Networking*. Pp 337-45 (2014).