Comparing Load Balancing With The Load Distribution For Different Networks

Harshit Nigam\textsuperscript{1}, Neeraj Verma\textsuperscript{2}, Gaurav Nigam\textsuperscript{3}, Vinay kumar pandey\textsuperscript{4}

Dept. of Computer Science and Engineering, \textsuperscript{1,2}United Institute Of Technology Allahabad,\textsuperscript{3}Lord Krishna college,\textsuperscript{4}S.P.Memorial Institute Of Technology, (U.P) India

\textsuperscript{1}contactharshitnigam@gmail.com, \textsuperscript{2}vermaneeraj13@gmail.com, \textsuperscript{3}gauravnigam250185@gmail.com, \textsuperscript{4}Vinaykprag85@gmail.com

Abstract— Due to the dramatically increase popularity of the services provided over the public interest, problem with current mechanisms for control and management of the internet and other networks built on the internet protocol suite do not provide sufficient support for the efficient control management of traffic, i.e for Traffic Engineering. In this paper comparison is made between load distribution and load balancing. Various method have been used to solve the problem and from the data packets point of view to reach to the destination avoiding to travel to the congested path or the path which had been failed due to network congestion but if we apply Dijkstra’s Algorithm to load Balancing then we will be getting appropriate path that to at a minimum cost.

Keywords—Load Balancing, Load Distribution, Comparison between Load balancing and load Distribution

I. INTRODUCTION

The past few years have seen an exponential increase in the use of two kinds of communication services. The first kind is Internet-based data service, such as www, e-mail, and pocketsize voice. The second is wireless mobile services, particularly circuit-based wireless voice. As the penetration of Internet-based data services increases, more people will demand high speed, wireless, bandwidth data services. Although current widely deployed wireless networks (e.g., GSM, CDMA), can provide short message service (SMS), they cannot meet the ever increasing bandwidth requirements of data services. The intermediate solution to this problem is the general packet radio service (GPRS)\textsuperscript{1}, because it provides higher speed data service to the end user, and relies on IP for core network transportation. However, the access network still adopts a conventional circuit-based network.

The inconsistency between the access network and core network causes many problems. One of these problems is the quality of service (QoS) support across the whole network.

In order to fully address these problems, 3GPP proposes the concept of an IP based access network \cite{2}. Recently, multi-protocol label switching (MPLS) has begun deployment in the Internet backbone to provide traffic engineering, which cannot be supported by the Conventional Internet \cite{4}. MPLS is also proposed as a transport option in the access network of next generation wireless networks \cite{5}. One of the critical problems associated with mobile networks is mobility management; thus, it is important to handle mobility management in an MPLS based access network.

Load balancing is used for distributing workloads across multiple computing resources, such as computers, network links, central processing units or disk drives. Load balancing goal is to optimize resource use, maximize throughput, minimize response time, and avoid overload of any one of the resources. Using multiple components with load balancing instead of a single component may increase reliability through redundancy. Load balancing is usually provided by dedicated software or hardware, such as a multilayer switch or a Domain Name System server process.\cite{7} One of the most commonly used applications of load balancing is to provide a single Internet service from multiple servers, sometimes known as a server farm. Like client and Server. Commonly, load-balanced systems include popular web sites, large Internet Relay Chat networks, high-bandwidth File Transfer Protocol sites, Network News Transfer Protocol and Domain Name System (DNS) servers.
For Internet services, the load balancer is usually a software program that is listening on the port where external clients connect to access services. It also prevents clients from contacting backend servers directly, which may have security benefits by hiding the structure of the internal network and preventing attacks on the kernel’s network stack or unrelated services running on other ports. The load balancer forwards requests to one of the "backend" servers, which usually replies to the load balancer. This allows the load balancer to reply to the client without the client ever knowing about the internal separation of functions. [9]

Some load balancers provide a mechanism for doing something special in the event that all backend servers are unavailable. This might include forwarding to a backup load balancer, or displaying a message regarding the outage. It can automatically provide the amount of capacity needed to respond to any increase or decrease of application traffic. It is also important that the load balancer itself does not become a single point of failure. Usually load balancers are implemented in high-availability pairs which may also replicate session persistence data if required by the specific application.

The technique has other advantages and disadvantages, depending on the degree of control over the DNS server and the granularity of load balancing desired.[2] This technique works particularly well where individual servers are spread geographically on the Internet.

Figure 1: Load Balancing according to different paths present

II. ROUND-ROBIN DNS

An alternate method of load balancing, which does not necessarily require a dedicated software or hardware node, is called round robin DNS. In this technique, multiple IP addresses are associated with a single domain name; clients are expected to choose which server to connect to. Unlike the use of a dedicated load balancer, this technique exposes to clients the existence of multiple backend servers.

Figure 2: Packets are delivered according to different weights presents

III. SCHEDULING ALGORITHM

A variety of scheduling algorithms are used by load balancers to determine which server to send a request to. Simple algorithms include random choice or round robin DNS. More sophisticated load balancers may take into account additional factors, such as a server's reported load, recent response times, up/down status (determined by a monitoring poll of some kind), number of active connections, geographic location, capabilities, or how much traffic it has recently been assigned. Number of clients increase or decrease within the geographical area. While the Number of times host come to form connection

Use in telecommunications

Load balancing can be useful in applications with redundant communications links. For example, a company may have multiple Internet connections ensuring network access if one of the connections fails.[4] This is done for security point of view.

A failover arrangement would mean that one link is designated for normal use, while the second link is used only if the primary link fails. To avoid network congestion.

Using load balancing, both links can be in use all the time. A device or program monitors the availability of all links and selects the path for sending packets.
Use of multiple links simultaneously increases the available bandwidth. Many telecommunications companies have multiple routes through their networks or to external networks. They use sophisticated load balancing to shift traffic from one path to another to avoid network congestion on any particular link, and sometimes to minimize the cost of transit across external networks or improve network reliability. By doing this alternate arrangement communication failure condition does not arises.

Another way of using load balancing is in network monitoring activities. Load balancers can be used to split huge data flows into several sub-flows and use several network analyzers, each reading a part of the original data. where complex processing of the data may not be possible at wire speed.

Load distribution seeks to improve the performance of a distributed system, usually in terms of response time or resource availability, by allocating workload amongst a set of cooperating hosts.

This division of system load can take place statically or dynamically:

Static load distribution assigns jobs to hosts deterministically, without consideration of runtime events. This approach is both simple and effective when the workload can be accurately characterized and where the scheduler is pervasive, in control of all activity, or is at least aware of a consistent background over which it makes its own distribution. Problems arise when the background load is liable to fluctuations, or there are jobs outside the control of the static load distributor.

Dynamic load distribution is designed to overcome the problems of unknown or workloads, non-pervasive scheduling and runtime variation the interaction of human beings can alter resource requirements or availability). Dynamic load distribution systems typically monitor the workload and hosts for any factors that may affect the choice of the most appropriate assignment and distribute jobs accordingly. This very difference between static and dynamic forms of load distribution, is the source of the power and interest in dynamic load distribution. [8]

The objectives of this thesis lie entirely within the domain of dynamic load balancing. For brevity, I will take the more general term of load distribution to stipulate only the dynamic form.

The Degree of Load Distribution

The essential objective of load distribution is to divide the workload amongst a cooperating group of hosts. This objective may be fulfilled with varying degrees of fineness.

Load distribution is usually described in the literature, as either load balancing or load sharing. These terms are often used interchangeably, but can also attract quite distinct definitions. I will adopt the two terms, and use them in the strictest sense to describe the degree to which workload is distributed, and introduce a third term to describe the middle ground.

The distinction between load leveling and load balancing schemes is more difficult. In particular, to meet the definition, a load balancing scheme must continue to redistribute load until it meets a balance criterion. The architecture of the system is important, as it can suggest the most appropriate degree of load distribution.

- **Load Sharing:** network of personal workstations would be prohibitively expensive to balance due to the overheads of load and state collection, yet the detection and utilization of idle workstations for load sharing is quite feasible as shown by the Butler system. This is the coarsest form of load distribution. Load may only be placed on idle hosts, and can be viewed as binary, where a host is either idle or busy.

- **Load Balancing:** Where load sharing is the coarsest form of load distribution, load balancing is the finest. Load balancing attempts to make sure that the workload on each host is within a small degree (or balance criterion) of the workload present on every other host in the system.

- **Load Leveling:** Load leveling occupies the ground between the two extremes of load sharing and load balancing. Rather than trying to obtain a strictly even distribution of load across all hosts, or simply utilizing idle hosts, load leveling seeks to avoid congestion on any one host.

Thus load sharing, leveling and balancing define a continuum from a course to a fine distribution of load, and seek to distinguish the sometimes unstated intentions of different load distribution schemes. [9]
IV. LOAD SPLITTING VERSUS LOAD BALANCING

Load splitting and load balancing are not the same. Load splitting provides a means to randomly distribute traffic streams across multiple equal-cost reverse path forwarding paths, which does not necessarily result in a balanced IP multicast traffic load on those equal-cost RPF paths.[12] By randomly distributing traffic streams, the methods used for load splitting IP multicast traffic attempt to distribute an equal amount of traffic flows on each of the available RPF paths not by counting the flows, but, rather, by making a pseudorandom decision. These methods are collectively referred to as ECMP multicast load splitting methods. ECMP multicast load splitting methods, thus, result in better load-sharing in networks where there are many traffic streams that utilize approximately the same amount of bandwidth. [11]

V. LOAD DISTRIBUTION ALGORITHM

1. Server-Initiated Algorithm

This activity initiated by an overloaded node(sender) that attempt to send a task to an under loaded node(receiver)

2. Receiver- Initiated Algorithm

The load distribution activity is initiated from under loaded node(receive) that is trying to obtain a task from an overloaded node(sender)

3. Symmetrically- Initiated Algorithm

Both sender and receiver searches for receivers and senders


VI. EQUAL-COST MULTIPATH (ECMP)

Increasing efficiency and quality demands of services from IP network service providers and end users drive developers to offer more and more sophisticated traffic engineering methods for network optimization and control. Intermediate System to Intermediate System and Open Shortest Path First are the standard routing solutions for intra-domain networks.

An easy upgrade utilizes Equal Cost Multipath (ECMP) that is one of the most general solutions for IP traffic engineering to increase load balancing and fast protection performance of single path interior gateway protocols. Equal-Cost Multipath (ECMP) is a mechanism for achieving load balancing for a path with multiple equal cost next-hops. By default, in the event of equal-cost next-hops Juniper will randomly pick one next-hop per route to install for hardware forwarding. There is also an option that allows multiple next-hops to be installed into hardware, known as "per-packet load balancing". The term "per-packet load balancing" as used by Juniper is actually misleading in two ways. First, only the first generation Internet Processor I ASIC based routers were capable of true per-packet load-balancing, which is generally not a desirable behavior. All other Juniper routers actually implement a "per-flow" load-balancing, which prevents packet reordering, but the configuration element is still called "per-packet" for compatibility. Second, this option doesn't actually have a direct relationship to "how" the packets are forwarded at all, it simply enables the installation of multiple next-hops for the same route. [8]

Figure 3: Load distribution using ECMP

VII. CONCLUSION

We had seen both load distribution and load balancing algorithms. There are some faults in Load distribution that is load is evenly distributed approxly but not actually. New path is chosen when an error occurred , in ECMP technique second path is chosen but in case of Load Balancing technique using Dijkstra Algorithm many path are present where load is balanced and dijkstra helps in choosing the path with minimum distance traveled as well as Round robin DNS is used after that the path selection is not appropriate at the time of failure or when the network congestion arrives.
In that we will choose the alternate route to reach to the destination but if we use the dijkstra's Algorithm we will be reaching the destination that to at a minimum cost.

REFERENCES

AUTHOR’S PROFILE
Harshit Nigam obtained Btech (HONOURS) in (Computer Science and Engineering) from G.B.T.U in 2011 and pursing Mtech (Computer Science) from United Institute of Technology Allahabad. He is working as a lecturer in S.P.Memorial Institute of Technology. His area of interest includes Information Security and Graph Theory. He has more than 2 year teaching experience in the area of interest algorithm.

Neeraj Verma was born in Uttar Pradesh, India. He graduated from Uttar Pradesh Technical University, Luck now, India in 2007. Now he is Assistant Professor in department of Computer Science and Engineering, Kamla Nehru Institute Of Technology , Sultanpur. He has more than 2 year teaching experience in the area of interest

Gaurav Nigam obtained his B.Tech Degree in ECE and completed M.Tech Microwave Engg from SHIATS Allahabad .He guided various projects at Engineering Level .His area of interest include Antenna ,Communication System and Solid State Devices

Vinay Kumar Pandey pursuing M.Tech( Computer Science) from Integral university and having an experience of more than 6 years.