RESEARCH ARTICLE OPEN ACCESS

Network Load Balancing and Its Performance Measures

Ms. Arti Mishra
Assistant Professor
Department of Computer Science and Engineering
Shri Ram Murti Smarak College of Engineering & Technology
SRMSWCET, Bareilly
Uttar Pradesh - India

ABSTRACT

Load balancing is a way to spread tasks out over multiple resources. By processing tasks and directing sessions on different servers, load balancing helps a network avoid annoying downtime and delivers optimal performance to users. There are virtual load balancing solutions that work in a manner similar to virtual applications or server environments. There are also physical load balancing hardware solutions that can be integrated with a network. The method used depends entirely upon the team implementing the solution and their particular needs. **Network Load Balancing (NLB)** is a clustering technology offered by Microsoft as part of all Server and Windows Server 2003 family operating. NLB uses a distributed algorithm to load balance network traffic across a number of hosts, helping to enhance the scalability and availability of mission critical, IP -based services, such as Web, virtual private networking, streaming media, terminal services, proxy and so on. It also provides high availability by detecting host failures and automatically redistributing traffic to operational hosts. This paper describes the detailed architecture of network load balancing, various types of addressing and the various performance measures.

Keywords:- Addressing, Load balancing, Network, performance**.**

I. INTRODUCTION

Network load balancing is an efficient and cost-effective solution designed to enhance the availability and scalability of Internet applications by allowing system administrators to build clusters, which are load balanced with incoming client requests. During NLB, clients cannot distinguish the cluster from a single server. Server programs are also unaware that a cluster is running.

As a result of this setup, NLB allows for greater overall control, including remote cluster management from any network point. Administrators can tailor clusters to services with port-defined controls. Cluster hosts and software may be modified without service interruption. NLB sends regular messages, allowing all cluster members to monitor the other hosts' presence. Host failures and recovery are handled automatically and quickly. NLB's software implementation requires extremely low overhead to handle network traffic. The process delivers excellent performance scaling, which is limited only by subnet bandwidth.

Network Load Balancing provides scalability and high availability to enterprise-wide TCP/IP services, such as Web, Terminal Services, proxy, Virtual Private Networking (VPN), and streaming media services. Network Load Balancing brings special value to enterprises deploying TCP/IP services, such as e-commerce applications, that link clients with transaction applications and back-end databases.

Network Load Balancing servers (also called *hosts*) in a cluster communicate among themselves to provide key benefits, including:

- **Scalability:** Network Load Balancing scales the performance of a server-based program, such as a Web server, by distributing its client requests across multiple servers within the cluster. As traffic increases, additional servers can be added to the cluster, with up to 32 servers possible in any one cluster.

- **High availability:** Network Load Balancing provides high availability by automatically detecting the failure of a server and repartitioning client traffic among the remaining servers within ten seconds, while providing users with continuous service.

  Network Load Balancing distributes IP traffic to multiple copies (or *instances*) of a TCP/IP service, such as a Web server, each running on a host within the cluster. Network Load Balancing transparently partitions the client requests among the hosts and lets the client's access the cluster using one or more "virtual" IP addresses [1]. From the client's point of view, the cluster appears to be a single server that answers these client requests. As enterprise traffic increases, network administrators can simply plug another server into the cluster.

  For example, the clustered hosts in Fig. 1 below work together to service network traffic from the Internet.

Each server runs a copy of an IP-based service, such as Internet Information Services 5.0 (IIS), and Network Load Balancing distributes the networking workload among them.

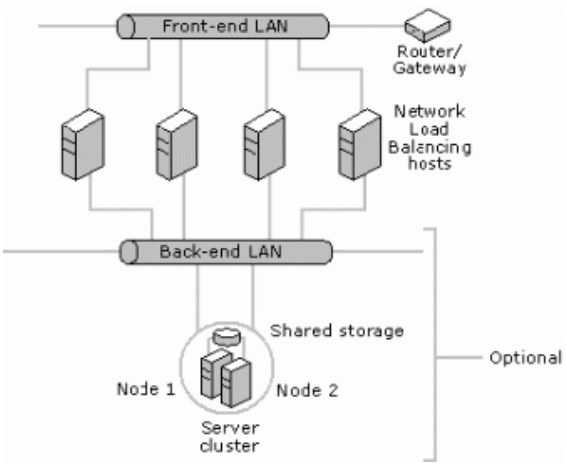

Fig. 1: A four-host cluster works as a single virtual server to handle network traffic. Each host runs its own copy of the server with Network Load Balancing distributing the work among the four hosts.

This speeds up normal processing so that Internet clients see faster turnaround on their requests. For added system availability, the back-end application (a database, for example) may operate on a two-node cluster running Cluster service.

II. NETWORK LOAD BALANCING CLUSTER

Network Load Balancing relies on the fact that incoming packets are directed to all cluster hosts and passed to the Network Load Balancing driver for filtering. Network Load Balancing cluster can configure in one of the following modes:

Multicast: Multicast mode allows communication among hosts because it adds a Layer 2 multicast address to the cluster instead of changing the cluster. Communication among hosts is possible because the hosts retain their original unique media access control (MAC) addresses and already have unique, dedicated IP addresses. However, the address resolution protocol (ARP) reply that is sent by a host in the cluster(in response to an ARP request) maps the cluster's unicast IP address to its multicast MAC address.

Unicast: Unicast mode works seamlessly with all routers and Layer 2 switches. However, this mode induces switch flooding, a condition in which all switch ports are flooded with Network Load Balancing traffic, even ports to which servers not involved in Network Load Balancing are attached. To communicate among hosts, you must have a second virtual adapter for each host.

III. NETWORK LOAD BALANCING ARCHITECTURE

Network Load Balancing runs as a network driver logically beneath higher-level application protocols, such as HTTP and FTP. On each cluster host, the driver acts as a filter between the network adapter's driver and the TCP/IP stack, allowing a portion of the incoming network traffic to be received by the host. This is how incoming client requests are partitioned and load-balanced among the cluster hosts. To maximize throughput and availability, Network Load Balancing uses fully distributed software architecture, and an identical copy of the Network Load Balancing driver that runs in parallel on each cluster host [2]. The fig.2 given below shows the implementation of Network Load Balancing as an intermediate driver in the Windows Server 2003 network stack.

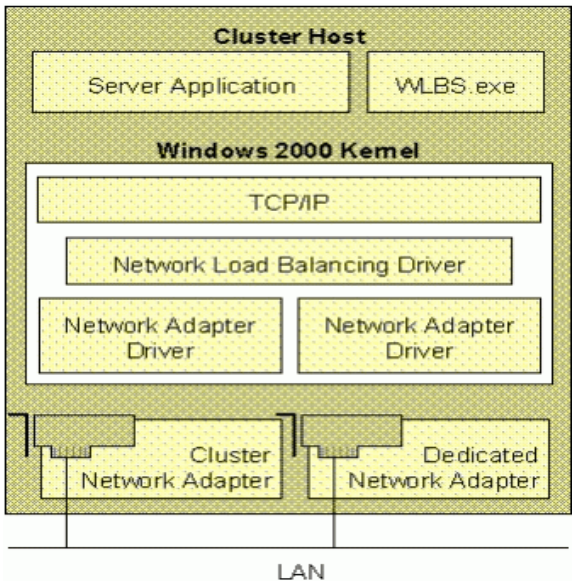

Fig. 2: Network Load Balancing runs as an intermediate driver between the TCP/IP protocol and network adapter drivers within the Windows 2000 protocol stack.

This architecture maximizes throughput by using the broadcast subnet to deliver incoming network traffic to all cluster hosts and by eliminating the need to route incoming packets to individual cluster hosts. Since filtering unwanted packets is faster than routing packets (which involves receiving, examining, rewriting, and resending), Network Load Balancing delivers higher network throughput than dispatcher-based solutions. As network and server speeds grow, its throughput also grows proportionally, thus eliminating any dependency on a particular hardware routing implementation. For example, Network Load Balancing has demonstrated 250 megabits per second (Mbps) throughput on Gigabit networks.

IV. NETWORK LOAD BALANCING ADDRESSING

The Network Load Balancing cluster is assigned a primary Internet Protocol (IP) address. This IP address represents a virtual IP address to which all of the cluster hosts respond, and the remote control program that is provided with Network Load Balancing uses this IP address to identify a target cluster.

Primary IP address

The primary IP address is the virtual IP address of the cluster and must be set identically for all hosts in the cluster. You can use the virtual IP address to address the cluster as a whole. The virtual IP address is also associated with the Internet name that you specify for the cluster.

Dedicated IP address

You can also assign each cluster host a dedicated IP address for network traffic that is designated for that particular host only. Network Load Balancing never load-balances the traffic for the dedicated IP addresses, it only load-balances incoming traffic from all IP addresses other than the dedicated IP address.

The following figure (Fig.3) shows how IP addresses are used to respond to client requests.

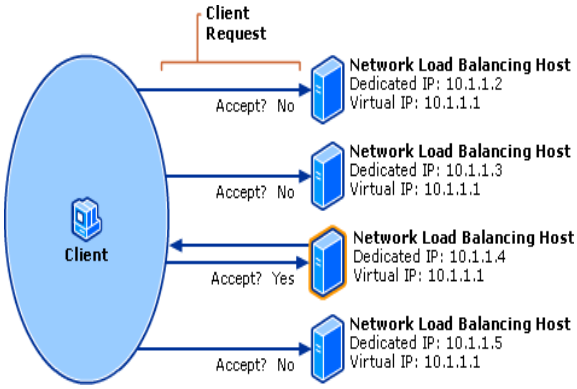

Fig.3: Network Load Balancing Cluster

Network Load balancing cluster hosts exchange heartbeat messages to maintain consistent data about the cluster's membership. By default, when a host fails to send out heartbeat messages within five seconds, it is deemed to have failed. Once a host has failed, the remaining hosts in the cluster perform convergence and do the following:

- Establish which hosts are still active members of the cluster.
- Elect the host with the highest priority as the new default host.
- Ensure that all new client requests are handled by the surviving hosts.

In convergence, surviving hosts look for consistent heartbeats. If the host that failed to send heartbeats once again provides heartbeats consistently, it rejoins the cluster in the course of convergence. When a new host attempts to join the cluster, it sends heartbeat messages that also trigger convergence. After all cluster hosts agree on the current cluster membership, the client load is redistributed to the remaining hosts, and convergence completes [3].

The following figure (Fig.4) shows how the client load is evenly distributed among four cluster hosts before convergence takes place:

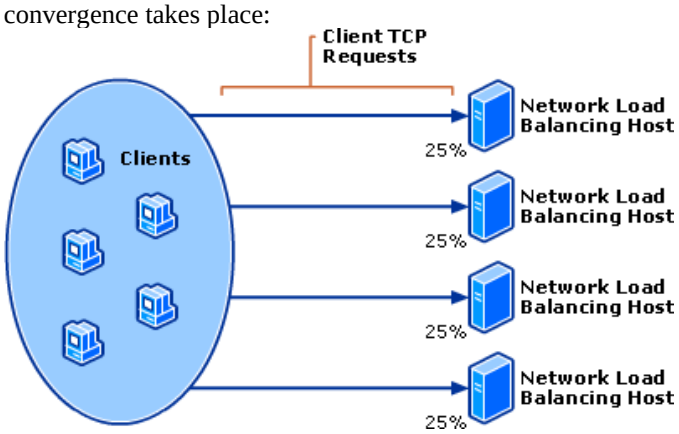

Fig.4: Network Load Balancing Cluster before Convergence

The following figure (Fig.5) shows a failed host and how the client load is redistributed among the three remaining hosts after convergence

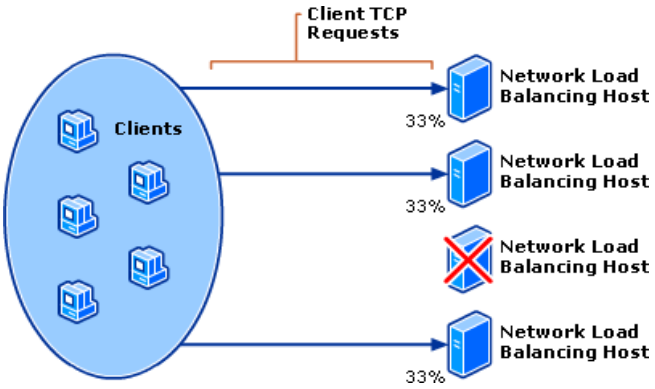

Fig.5: Network Load Balancing Cluster after Convergence

Convergence generally only takes a few seconds, so interruption in client service by the cluster is minimal. During convergence, hosts that are still active continue handling client requests without affecting existing connections [4]. Convergence ends when all hosts report a consistent view of the cluster membership and distribution map for several heartbeat periods.

By editing the registry, you can change both the number of missed messages required to start convergence and the period between heartbeats. However, be aware that making the period between heartbeats too short increases network overhead on the system. Also be aware that reducing the number of missed messages increases the risk of erroneous host evictions from the cluster.

Selecting an IP Transmission Mode

There is no restriction on the number of network adapters, and different hosts can have a different number of adapters. You can configure [5] Network Load Balancing to use one of four different models.

Single Network Adapter in Unicast Mode:

The single network adapter in unicast mode is suitable for a cluster in which you do not require ordinary network communication among cluster hosts, and in which there is limited dedicated traffic from outside the cluster subnet to specific cluster hosts. In this model, the computer can also handle traffic from inside the subnet if the IP datagrams do not carry the same MAC address as the cluster adapter.

Single Network Adapter in Multicast Mode

This model is suitable for a cluster in which ordinary network communication among cluster hosts is necessary or desirable, but in which there is limited dedicated traffic from outside the cluster subnet to specific cluster hosts.

Multiple Network Adapter in Unicast Mode

This model is suitable for a cluster in which ordinary network communication among cluster hosts is necessary or desirable, and in which there is comparatively heavy dedicated traffic from outside the cluster subnet to specific cluster hosts. This mode is the preferred configuration used by most sites because a second network adapter may enhance overall network performance.

Multiple Network Adapter in Multicast Mode

This model is suitable for a cluster in which ordinary network communication among cluster hosts is necessary, and in which there is heavy dedicated traffic from outside the cluster subnet to specific cluster hosts.

The advantages and disadvantages of each model are listed in the following table given below (Table1).

Table 1: Comparison of Modes

Adapter	Mode	Advantages	Disadvantages
Single	Unicast	Simple configuration	Poor overall performance
Single	Multicast	Medium performance	Complex configuration
Multiple	Unicast	Best balance	None
Multiple	Multicast	Best balance	Complex configuration

V. NETWORK LOAD BALANCING PERFORMANCE

The performance impact of Network Load Balancing can be measured in four key areas:

- **CPU overhead** on the cluster hosts, which is the CPU percentage required to analyze and filter network packets (lower is better).
- **Response time** to clients, which increases with the non-overlapped portion of CPU overhead, called *latency* (lower is better).
- **Throughput** to clients, which increases with additional client traffic that the cluster can handle prior to saturating the cluster hosts (higher is better).
- **Switch occupancy**, which increases with additional client traffic (lower is better) and must not adversely affect port bandwidth.

In addition, Network Load Balancing scalability determines how its performance improves as hosts are added to the cluster. Scalable performance requires that CPU overhead and latency not grow faster than the number of hosts.

VI. CONCLUSION

Network Load Balancing is superior to other software solutions such as round robin DNS (RRDNS), which distributes workload among multiple servers but does not provide a mechanism for server availability. If a server within the host fails, RRDNS, unlike Network Load Balancing, will continue to send it work until a network administrator detects the failure and removes the server from the DNS address list. This results in service disruption for clients. Network Load Balancing also has advantages over other load balancing

solutions—both hardware- and software-based—that introduce single points of failure or performance bottlenecks by using a centralized dispatcher. Because Network Load Balancing has no proprietary hardware requirements, any industry-standard compatible computer can be used. This provides significant cost savings when compared to proprietary hardware load balancing solutions.

REFERENCES

[1] M. Andreolini, M. Colajanni and R. Morselli, "Performance Study of Dispatching Algorithms in Multi-tier Web Architectures," Performance Evaluation Review, Vol. 30, No. 22, pp.10-20, Sept. 2002.

[2] T. Desell, K.E. Maghraoui and A. C. Varel,, "Load Balancing of Autonomous Actors over Dynamic Networks," Proceedings Hawaii International Conference on System Sciences, Track 9, Vol. 9, Page 90268.1, 2004.

[3] Li Wenzheng, Shi Hongyan: "Novel Algorithm for Load Balancing in Cluster Systems" Publidhe by IEEE Proceedings ofthe 2010-978-1-4244-6763-1/10.

[4] Remi Badonnel, Mark Burgess: "Dynamic Pull-Based Load Balancing for Autonomic Servers" Published by IEEE 978-1-4244-2066-7/08.

[5] Dhakal S. , "On the optimization of load balancing in distributed networks in the presence of delay, Advances in Communication Control Networks," LNCSE vol. 308, pp. 223–244,Springer-Verlag, 2004.

WEB REFERENCES

[1] http://msdn.microsoft.com/en-us/library/bb742455.aspx.

[2].http://en.wikipedia.org/wiki/Network_Load_Balancing_Services.

[3].http://technet.microsoft.com/enus/library/cc756878(v=ws.10).aspx