RESEARCH ARTICLE OPEN ACCESS

Single System Image Server Cluster

Shruti jawanjal*
*(Department of Computer Science, sant gadge baba university, Maharashtra)

ABSTRACT
The paper reports an implementation of Single System Image Server Cluster (SSISC). The Servers have been configured with same MAC and IP addresses and are connected through a two layer switch. Implementation has been done by modifying iptables and recompiling the linux kernel. This implementation has been tested for performance using WebStone. WebStone is a highly- configurable client-server benchmark for HTTP servers. This implementation removes a possibility of single point failure of web cluster. The implementation automatically detects newly added and failed server and hence reliable. The test indicates that requests are distributed to servers in the cluster, depends on the policy of distribution.
***Keywords* –** MAC, IP, iptables, recompiling, Webstone.

I. INTRODUCTION

A common technique used to build a high capacity web server is to use a cluster of server machines with a centralized dispatcher. The dispatcher receives all incoming requests and distributes them to one of the servers

II. Materials & Methods

2.1 Methods Used

All machines in a cluster are having with the same IP and MAC address, to avoid the need for router configuration in ONE-IP. This design is called the "Clone Cluster" [5] in this method all machines in the cluster are perfect images. All machines are attached to a shared-medium Ethernet LAN using simple 2 layer Enterasys VH-2402S2 switch [2] so that all packets are seen by all machines. The Linux kernel of each machine is modified and recompiled. This modification is done to provide machines with same MAC address with switch. The Clone Cluster configuration (Fig 1) is the same as ONE-IP, but no changes in the router are required. This method automatically detects newly added server. Failure of any server does not affect the system, so definitely the reliability is improved. Connection is established randomly with any machine in clone cluster.

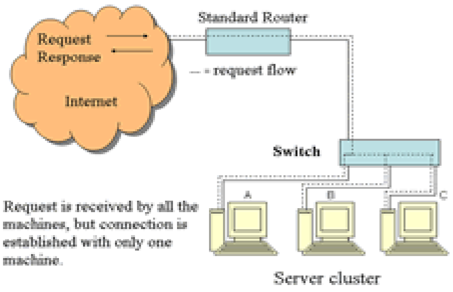

Fig 1 : Configuration for Clone Cluster

2.2 Server Configuration with two layer switch

The cluster machines are attached to a two layer switch [4]. As two-layer switches are needed of having different source MAC address on all of that switch port. To defend this conflict the source MAC address for the packets which are outgoing and present in Ethernet header are changed to the different MAC address then the kernel for each machine just because of this its recompilation and installation is done. This thing avoid the switch by getting the image of MAC address because if the MAC address is duplicated then all the all the packets will not delivered properly on switch port. The packets which are incoming present in Ethernet header they have same MAC address which is as of destination address and the packets which are outgoing they also present in the Ethernet header but they have MAC address of virtual machine which is as of source address. The switch changes its MAC vs. port table by keeping source address in Ethernet header. Thus switch not having clustered's copied MAC address. The client requires cluster's copied MAC address for communication. The client form an ARP [6][5] request to get cluster's copied MAC address. The structure of ARP packet is shown in figure 2. The ARP request packet is simple. ARP has source address which is client MAC address and destination address as broadcast address which is in the Ethernet header. It also has source address as client MAC address and IP address and in ARP header it has clusters IP header and empty field for MAC address.The ARP reply packet from cluster needs updating to avoid switch from getting cluster's actual MAC address. The ARP header contains cluster's IP and copied MAC address as source address & client's MAC and IP address as destination address. The Ethernet header in ARP reply packet contains clients MAC address as destination address and cluster machine's virtual MAC address as source address. Thus the difference in source MAC address in Ethernet and ARP header prevents the switch from getting cluster's copied MAC address contained in ARP header and at the same time provide cluster's copied MAC address to the client that is contained in ARP header. The switch gets the virtual MAC address of each machine.

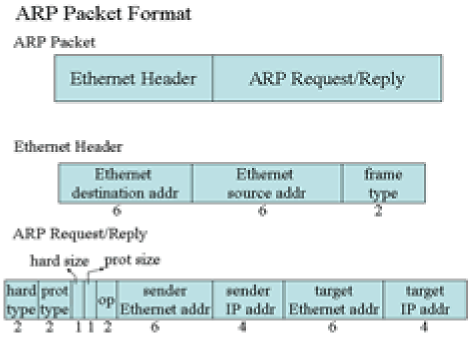

Fig 2: Format of ARP request or reply packet

III. Implementation

3.1 Implementing duplicate addresses

All standard Ethernet adapters contain a different burned-in MAC address. This address can be over-ridden by a locally administered address. Linux shell commands [6] can be used to assign a locally administered address.

The Ethernet cards on different machines are having with the same MAC address and attach the same secondary IP address to these interfaces. Before a host attaches a new IP address to its Ethernet card, it checks that no other host on the same LAN is using that IP address. If same IP address is found, both machines are informed and warnings are issued. When there are two machines with the same primary IP address, ARP detects this inconsistency and issues warning. A ARP packet is sent during initialization. Gratuitous ARP is an ARP packet with destination as broadcast address and the source address as the IP address of the machine. Its basic purpose is to announce the IP address of the new machine on the LAN. If there is another machine with the same IP already existing on the LAN then it comes to know of this machine using the same IP and issue a warning. To avoid detection of the duplicate IP address in the Clone Cluster, the ifup file in /sbin directory is modified. This modification prevents any action on detection of duplicate IP.

3.2 TCP connection

The transport layer provides a flow of data between two hosts, for the application layer. In TCP/IP protocol suite there are two vastly different transport protocols; TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). TCP is a connection-oriented protocol. The three-way TCP handshake (fig 5) completes the connection establishment. The requesting end (normally called the client) sends a SYN segment.

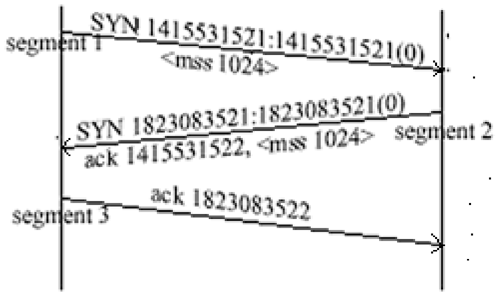

Fig3: TCP three-way handshake

IV. Results and Discussion

4.1 Evaluation Experiments

Two experiments were done to know the performance of the Clone Cluster using three load balancing policies.

Baseline experiment**:** This experiment baselines the performance of a single server machine. The Web Stone benchmark was used. Response time and server throughput were measured.

Two machine cluster experiment**:** This experiment measures the improvement (over a single machine) in response time and server throughput for a two machine Clone Cluster. The experimental set-up was the same as for the baseline experiment.

4.2 Evaluation Results

Figure 4 show the results for the one (baseline) and figure 5 & 6 for the two machine cluster experiments. The measurements are taken from Web Stone run log files. Figure 4 shows that the server throughput and average response time for one machine plateaus at about 150 clients. Further increases in load increases average response time and server throughput. This shows that the satisfaction point for the cluster has been reached. Figure 8 shows the server throughput and average response time for two machine cluster by using the policy to distribute the client requests by load. Similarly, figure 6 shows the server throughput and average response time for two machine cluster for random distribution of client requests. The two machine cluster removes the saturation point for one machine cluster.

Servers : A		
Clients	Server thruput Mbit/sec	Average response time sec
10	19.15	.009
30	20.67	.024
50	20.93	.040
70	19.93	.059
90	20.76	.073
110	20.80	.089
130	17.59	.124
140	20.43	.115
150	7.46	.332
170	4.26	.635
200	1.92	1.457

Fig4: performance analysis baseline experiment

Servers : A & B		Policy : byLoad
Clients	Server thruput Mbit/sec	Average response time sec
10	17.22	.009
30	17.01	.028
50	18.12	.044
70	17.45	.065
90	16.97	.086
110	16.98	.105
130	17.19	.123
140	17.03	.133
150	17.64	.138
170	16.93	.163
200	12.65	.254

Fig5: performance analysis of two machine experiment (by load policy)

Servers : A & B		Policy : byRandom
Clients	Server thruput Mbit/sec	Average response time sec
10	17.22	.009
30	17.53	.028
50	16.54	.050
70	17.95	.064
90	17.75	.084
110	18.21	.100
130	17.72	.121
140	17.59	.131
150	17.91	.138
170	17.54	.160
200	15.86	.206

Fig6: Performance analysis of two machine experiment (by random policy)

Fig6 shows the average response time vs. number of clients, graph for one and two machine image by distributing the client requests randomly and by load. It shows the satisfaction point for one machine cluster at 150 clients. This satisfaction point is passed in two machine cluster. The by Load and by Random policies show same in behavior. Figure 8 shows the server throughput time vs number of client's graph. The initial performance of one machine cluster is better than two machine cluster unto its saturation point. The two machine cluster shows consistent performance.

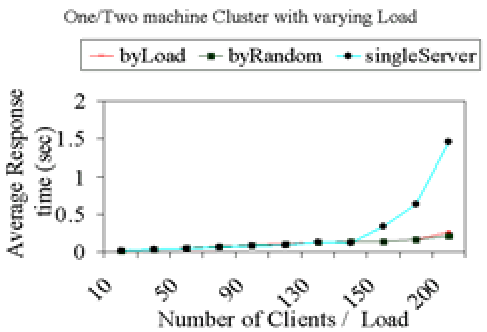

Fig 7: Average Response Time Vs Number of Clients

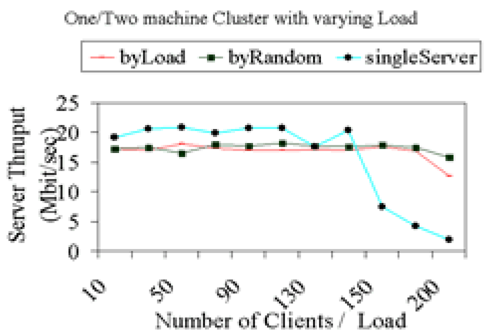

Fig8 : Server Throughput Vs Number of Clients

V. Acknowledgements

I would like to acknowledge sant gadge baba university Maharashtar, India to provide the facility.

References

[1]. T. Brisco, DNS Support for Load Balancing, RFC 1794, April 1995.

[2]. K. Egevang and P. Francis, ``*The IP Network Address Translator (NAT)*", RFC 1631, May 1994.

[3]. A. Bestavros, M. Crovella, J. Liu and D. Martin**,** *Distributed Packet Rewriting and its Application to Scalable Server Architectures*, Proceedings of the Sixth International Conference on Network Protocols, 1998.

[4]. O. Damani, P. Emerald Chung, Y. Huang, C. Kintala, and Y. Wang, "*ONE-IP: Techniques for Hosting a Service on a Cluster of Machines*", Journal of Computer Networks and ISDN Systems, Vol. 29, No. 8-13, pp. 1019-1027, 1997.

[5]. Sujit Vaidya and Kenneth J. Christensen, "*A Single System Image Server Cluster using Duplicated MAC and IP Addresses*", Proceedings of the 26th Annual IEEE Conference on Local Computer Networks**,** 2001.

[6]. W. Stevens, "*TCP/IP Illustrated Vol 1*", Addison Wesley Longman, 1998.