Load Balancing Program for Network Services

Nguyen Thi Thu Du, Tran Van Lang and Le Phuoc Loc

Institute of Applied Mechanics and Informatics, Vietnam Academy of Science and Technology, Vietnam

{ntthudu, tvlang, loc}@vast-hcm.ac.vn

Abstract— **ISP-servers system usually consist of many computers is strongly configured, and every computer often assumes a special function. This system often have some problems which are if one of servers has any breakdown, network services running on this server will not run, and some of servers which undertake important network services will run over load while others are still having many resources (i.e., CPU, memory etc), but cannot share. Load balancing program for network services is a new load balancing program for ISP-servers system.**

Index Terms—**Network, Grid Computing, Distributed Computing and Load Balance**

I. INTRODUCTION

LOAD balancing is the process of distributing some load across multiple links, servers, processors, or other devices in order to repair overload, improve performance and overcome deficiencies in existing equipment. Nowadays there are a lot of common load balancing methods such as the sever load balancing, DNS-based load balancing, firewall load balancing, and global server load balancing.

- Server Load Balancing (SLB) is a process that distributes traffic of a site among several servers using a network-based device.

- DNS-Based Load Balancing is based on technology of SLB. DNS operates also "round robin" principle and uses a function of DNS that allows more than one IP address to associate with a hostname.

- Firewall Load Balancing (FWLB) has been developed to get the better of the limitations of firewall technologies. Most firewalls are CPU-based, such as a SPARC machine or an x86-based machine. Because the processor is limited, the amount of throughput a firewall is often limited, too. Processor speed, packet size, configuration, and several other metrics are determining factors for what a firewall can do, but generally, they tend to max out at around 70 to 80 Mbps (Megabits per second) of throughput. Like as SLB, FWLB allows for the implementation of several firewalls sharing the traffic in a manner. However, the configuration and technology are different.

- Global Server Load Balancing (GSLB) implements have historically been based on DNS and SLB, but it distributes load to various locations as opposed to one location. DNS and SLB work on the Local Area Network (LAN), while GSLB works on the Wide Area Network (WAN). Better GSLB systems use advanced techniques for dealing with persistence on a global scale over simple DNS such as HTTP redirection and more advanced use global triangulation so URL is not changed. There are several ways to implement GSLB, such as DNS-based and BGP-based (Border Gateway Protocol). GSLB provides against data in case any site fails such as a fiber cut, a power outage, an equipment failure, or a meteor from outer space. There are limitations to both the DNS and BGP-based methods in GSLB. DNS-based GSLB is no guarantee that all of the West Coast users will be directed to the West Coast, or all of the East Coast users will be directed to the East Coast, and that everyone will be directed to another site in the event of a site-wide failure.

- Vendors are currently working on solutions. Though not 100 percent effective, GSLB is still an important technology and is employed by many large-scale sites.

Load balancing for a parallel system is an important problem that needs to be solved in order to enable the efficient use of parallel computer systems. Load balancing is often based on information using situation of system resources. In Grid computing system, the monitoring and discovery system component of Globus Toolkit manages and provides these information, as well as performance of current system. To solve the load balancing problem of mail server using SMTP protocol in the Grid computing system, we can use this component.

In this paper, we introduce into the MDS (Monitoring and Discovery System) of Globus Toolkit. And then, we explain technology solutions for building load balancing program, as well as deploy experiment with sending and receiving mail using the SMTP protocol on our IOIT-HCM Grid. The load balancing program is based on principle getting information about resource of all servers in Grid computing system using MDS (Monitoring and Discovery System) and selected the "freest" server to implement requests from clients. This program includes a server, which runs load balancer application, is set on transmission between selected server and
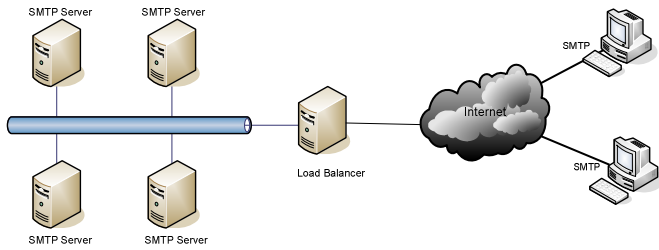
Fig. 1. Model of Load Balancing Program for SMTP server

outside client. Then, role of load balancer is a virtual server communicated directly with client and selected server. The purpose of load balancing program is data-way distribution of any network services to servers, which undertake correlation of same network service in SLB (Server Load Balancing) system.

The rest of paper was organized as follows. The section 2 briefly reviews the monitoring and discovery system. In the section 3, the load balancing program and bioinformatics application were described. Then, we present some conclusions in the section 4.

II.   MDS COMPONENT

Grid Information Services (GIS) is an important component in Grid Computing system. This component manages and provides information about status of a computational grid and all its components such as networks, compute nodes, storage systems, and instruments. The information is used to make decisions on distributing jobs in Grid environment [1].

MDS provides the necessary tools to build an LDAP-based information infrastructure for computational grids. MDS uses the LDAP protocol as a uniform means of querying system information from a rich variety of system components, and for optionally constructing a uniform namespace for resource information across a system that may involve many organizations. Therefore MDS defines an approach to present data to users (via the LDAP protocols and some particular schema), and some particular types of servers such as GRIS and GIIS being part of the Globus Toolkit. MDS can access to static and dynamic information of resources. Basically, it contains the following components:

− Information Provider
− Grid Resource Information Service (GRIS)
− Grid Index Information Service (GIIS)
− MDS client

The information providers translate the properties and status of local resources to the format defined in the schema and configuration files. In order to add new resource be used by MDS, the specific information providers have to be created to translate the properties and status to GRIS. The provided information must be in a format that GRIS understands.

MDS provides information sources of static host information (operating system version, CPU type, number of processors, etc.), dynamic host information (load average, queue entries, etc.), storage system information (available disk space, total disk space, etc.), as well as network information

via the Network Weather Service (network bandwidth and latency, both measured and predicted). From that we can use information about CPU to evaluate "free" level of computers in system [2].

III.   LOAD BALANCING PROGRAM

A.   Overview of Program

The common load balancing methods in the world have a limitation that is operated on "round robin" principle. To overcome this limitation, this paper had built load balancing program based on principle get resource information of all servers in Grid Computing system using MDS and selected the "freest" server to implement requests from clients.

The Fig. 1 is model of Load Balancing Program for SMTP server.

Client need only know Load Balancer server. When a client, is one of SMTP servers on Internet, execute a SMTP transaction, client will connect to Load Balancer server. Load Balancer server always listens to connections from clients. As soon as a client connect to Load Balancer server, and Load Balancer server will query to the system to get information about SMTP servers using MDS and then select the "freest" SMTP server in the cluster in the Grid Computing system and execute requests from this client.
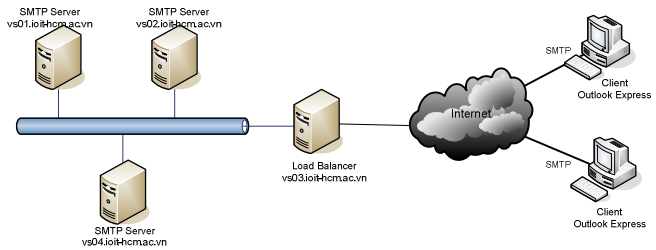

Fig. 2. Experiment of Load Balancing Program for SMTP service

B.   Technology Solution

Communication between Load Balancer server and clients must be right with SMTP (or ESMTP) protocol. That is, Load Balancer server always receives connections from clients at the port 25, and messages must conform to standard RFC 821, in 1869 [3].

There is a problem that is how the Load Balancer server could send these messages to servers inside. This sending can be implemented in IP layer or higher layer that is TCP layer (SMTP uses TCP protocol), even in application layer (specific is SMTP).

The communication shouldn't implement in application layer because Load Balancer server will be over loaded instead of servers inside.

If system was implemented in TCP layer, Load Balancer server has to listen requests to clients at a specific port (e.g., port 25 of SMTP protocol). When the Load Balancer server receives a request from client, it will create a connection to "freest" server and send data received from the client to the server. In this way, this application doesn't need to consider

session management of SMTP. However, Load Balancer server must be reserved two ports for a connection (the first port is used to connect to client and the second is used to connect to server).

Else if it was implements in IP layer, we don't have to consider limit to number of ports. Moreover, we can design a unique application for different services (SMTP, HTTP,). However, in this way, the application must intervene in operation system and must manage sessions

Based on these features, we decided to perform the transition on TCP layer.

IV. EXPERIMENTAL RESULTS

A. Mail Services

Configure of Load balancing program:

- Mail Servers:
    - SMTP Server: Postfix
    - POP3 Server: Cyrus-imapd
- Mail Client: Outlook Express

Because of deployment condition, this system uses port 5555 for SMTP service in Load Balancer server and implements connections from SMTP service to Load Balancer server using Outlook Express. This example is used to illustrate for connections, which send an e-mail from SMTP of clients outside to Load Balancer server. And then, Load Balancer server selects one of servers and implements requests from clients.

Connection Diagram Fig. 2, and configuration of client in Fig. 3(a), Fig. 3(b).

When a client use Outlook Express to send an e-mail to Load Balancer server using SMTP protocol, Load Balancer server will query to system in order to get information using MDS and then select the "freest" server in the cluster in the Grid Computing system and implement requests from this client.

- Incoming mail (POP3): is IP address of POP3 Server
- Outgoing mail (SMTP): is IP address of Load Balancer server
- Account name and password: are user and password registered with POP3 Server
- Server port numbers: is port, which Load Balancer server is listening.

Results: Load Balancer server Program is executed with 2 parameters in syntax:

```
java LoadBalancer <-p GATEWAYPORT>
[-ts TIMESTAMP]
```

- GATEWAYPORT: is port, which Load Balancing server is listening to Clients.
- TIMESTAMP: is time interval, which is used to get cpu-free value from MDS.
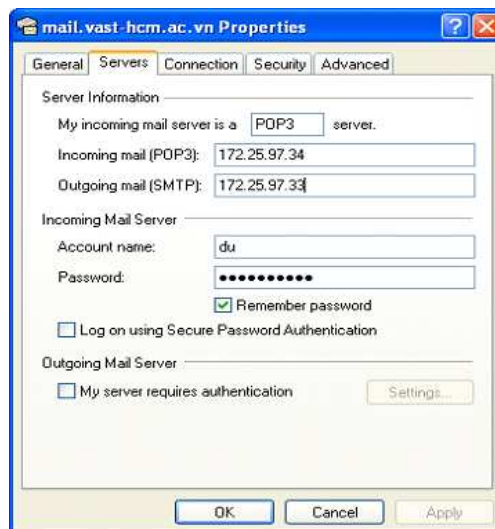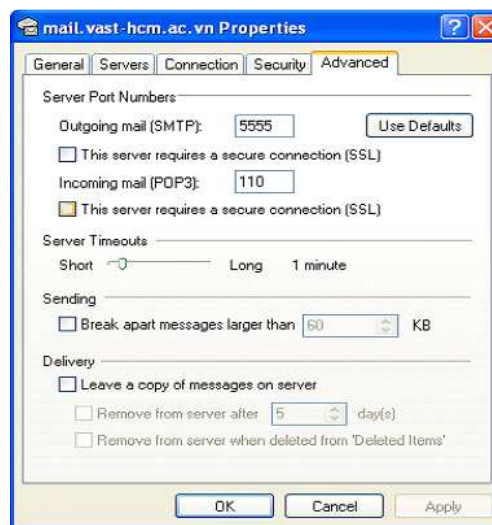

Fig. 3(a). Configuration of Client

Fig. 3(b). Configuration of Client

- TIMESTAMP=1 (default) : Mds-Cpu-Total-Free-1minX100

- TIMESTAMP=5: Mds-Cpu-Total-Free-5minX100

- TIMESTAMP=15: Mds-Cpu-Total-Free-15minX100

Time interval represents opinion of administrators and it is based on "free" level of servers. This value is chosen depending on short for using of servers in the place deployed. If servers have many changes on status in a short time interval, we use the option TIMESTAMP=1. Else if servers have a few changes, we use the option TIMESTAMP=15, (Results of Load Balancing Program in Fig. 4).

A. Motif finding Application

To illustrate load balancing ability of program, we used client/server applications. We deployed experiment with Motif finding application [4]. Run server application on all of computers, which are used for load balancing to listen and implement requests from clients. When a client make a

Fig. 4. Result of Load Balancing Program

connection, Load Balancer server queries to system to get information from MDS and select the "freest" computer in the cluster in the Grid Computing system and implement requests of the client.

We experiment with steps following:

- Send 300 (400, 500) requests from clients to Load Balancer server at the same time
- Calculate running time on 1 computer (2 computers)

These results indicate that Load Balancing Program execute fairly good. Running time interval on two computers is only approximate haft of running time interval on a single computer (This result is illustrated by chart in Fig. 5).
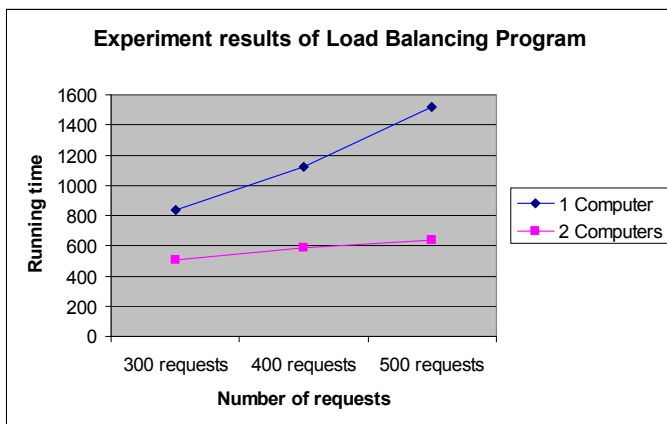

Fig. 5. Finding the motif with Load Balancing program was resident

Number of using sequences was given in Table 1.

Table 1. Sequences for Motif Finding

No. of Computer	300 requests	400 requests	500 requests
1	840s	1120s	1522s
2	508s	586s	640s

V. CONCLUSION

Requirements, that permit of load balancing client/server services using TCP/IP protocol and state information's servers is gotten from MDS component in Globus-Toolkit and selected the "freest" computer to implement requests from clients, has been well satisfied in our load balancing program. Load balancing program is experimented with sending and receiving mail using the SMTP protocol on small scale (in Institute of Applied Mechanics and Informatics).

Besides, on a distributed system such as cluster computing, grid computing, cloud computing as well as sky computing, the load balancing is always been concerned to optimize the activating of services.

REFERENCES

[1] Tran Van Lang, *Grid computing: building computing system and deploying applications* (in Vietnamese), Vietnam Education Publishing House, 2008, 196p.

[2] Bart Jacob, et al, *Enabling Applications for Grid Computing with Globus* - IBM Redbooks 2003, 140-144

[3] Tony Bourke, *Server Load Balancing*, O'REILLY' (August 2001), 14-23

[4] Tran Van Lang, *Using IT methods for solving molecular biology problems* (in Vietnam), Vietnam Education Publishing House, 2008, 230p.