Volume 4, Issue 9, September 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Result Paper on Secured Load Balancing Algorithm in Peer to Peer Network

[1]Megha M. Nalgirkar, [2]Prof. N. D. Kale

[1]M.E.(II year),Departmentof Computer Engineering,PVPIT,Bavdhan,Pune,Maharashtra, India.

[2]AssistantProfessor,Departmentof ComputerEngineering,PVPIT,Bavdhan,Pune, Maharashtra, India.

Abstract*—Due to lack of centralized control, the Distributed system and Peer-to-Peer Network System remain unsuccessful. All the loads are distributed to different nodes according to their functional performance. The non static load balancing is a topic used in distributed networking system. This paper analyses various security vulnerability for load balancing mechanism. In this paper we have proposed an algorithm which gives security and good functional and efficient performance to nodes. This Security load balancing algorithm (SLBA) has two elementary tasks. First, it achieves load balance to all nodes and second, it gives security by generating ID in load transfer in between the nodes. The algorithm is results in process the external load function to random nodes.

***Index Terms*— Security, Load balancing, p2p system, Distributed hash table, Load balancing, Load reduction.**

I. INTRODUCTION

Peer to peer system is combined solution to sharing and locating different resources over networking. If our system is homogenous i.e. all nodes have same resources then it is difficult to give load to all nodes then it is important to take heterogeneous system but when nodes not system are heterogeneous, the load assigned to whole system the nodes get heavy loaded, so by using this security load balancing (SLB) we can make our networking system more reliable and easily available[10]. When we using this distributed system then we have very limited no. balancing load. Otherwise many nodes are heavily loaded. So we can apply SLB which results in significant improvement in computing efficiency[1].

Basic thing in load balancing is we must have two types of nodes. One is heavily loaded and another is lightly loaded. Now we can combine both nodes respectively. Loads are redistributing among two nodes. But when the system is globally situated, where no. of nodes is heavily loaded and no. of Nodes are lightly then there arises of issue that: " How to recognize different nodes are heavily and lightly loaded?" This problem can be ironed out by forwarding queries by each node to other nodes. If the redistribute the load among them use it is average loaded. But it gave sometime wrong output and time consumable. In the existing system of load balancing we have schedules with predetermined node, but in SLB we can make schedule during runtime [2]. SLB mechanism has all information about the nodes; this information may be local or global. In this paper, we inspect safety vulnerability of DTH load balancing result. We suggest SLBA, a secured load balancing algorithm for DTH which carry heterogeneity networking with security.

A. Servers and sub servers:

Load balancer always tries to share the server load, whenever their load exceeds. From client side server has no.s of request is important aim. Servers can share these load function among the nodes. But due to unpredicted load performance of server is collapse. To overcome such problems we proposed secured algorithm in which server allocated the work to sub server (proxy server)[1]. System model for load balancing using servers and sub servers is shown in fig1.

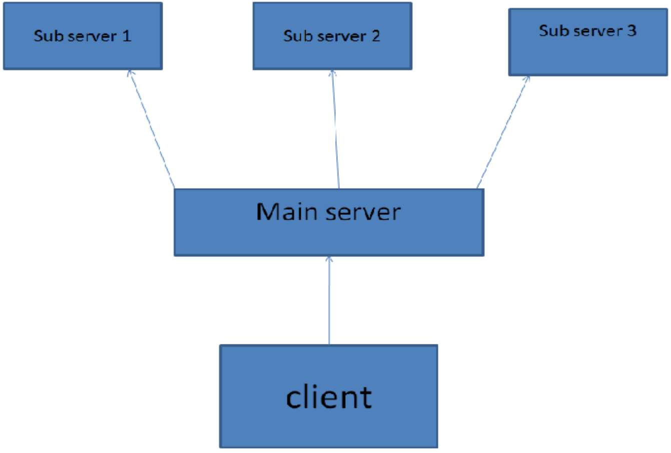


Fig 1: System model for load balancing using servers and sub servers

B. ID manipulating load balancing :

If each and every node have its own ID then at the time of reassigning load to node when nodes are join or leave then change of ID may degrade the system. For solution to this problem is virtual server assign thousand of position to ID space to each node, it will choose only one to whom virtual nodes will be in working situation[1]. Node's load balancing by ID manipulation and virtual servers are shown in fig2.

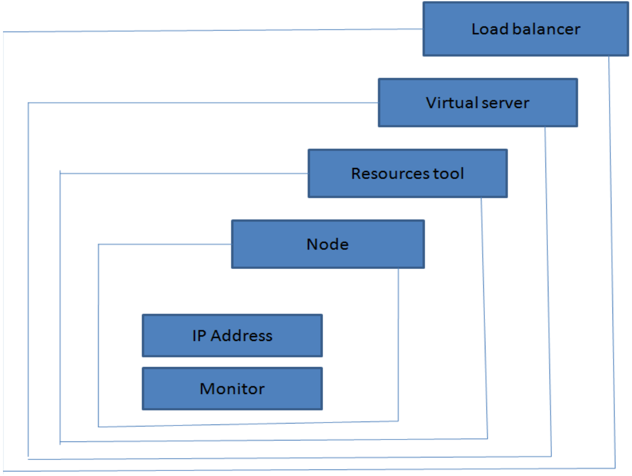


Fig 2: Load balancing structure by ID Manipulation and Virtual Server

C. Load balancing by virtual server:

We use the terminology 'Virtualization' for hardware load on system of server. It does function more efficient. These virtual servers are allowed to reduce the actual load on server and server reduces this load on nodes[1]. Virtual server are used in many concepts where continuously loads are redistributed due to nodes get exit, join and transfer. By using Virtual servers to load balancing of node all the loads to nodes are equally distributed.

D. Objectives of load balancing Technique

- Evaluate different features of reducing load of server.
- Recommending a new algorithm for overloaded targeted nodes.
- Examine, design and discover new solution to load balancing problem.
- Evaluate the distinct unit of servers and its sub servers[8].

II. LITERATURE SURVEY

Load balancing terminology in heterogeneous as well as homogenous networking system is widely studied in distributed system across the world. The basics of load balancing is that suppose we consider two nodes having random loads over them and external node is triggered with new join node. This newly joined node attempt to find highest load on two of old node and share it. Same concept is assigned to node if it wants to leave the system, it search for lightly loaded node and handover such load to it and exit. Such an existing load balancing system over the internet is not fully secured so many researchers find different technique.

i. **Histogram-based load balancing structure:**

In this technique we can propose a new framework known as histogram based load balancing which make possible reduced load balancing structure in peer to peer system. In this process we have histogram manger which preserve histogram to show the distribution of load among the nodes[5]. It also show the nodes are lightly loaded or heavily loaded, overloaded or under loaded. Another component in this technique is load balancing manager which is responsible to redistribute the load from overload node to under load and viceversa. Fig3 shows different nodes with nonoverlapping group. Each node is connected with its associate node which share the loads among them which is shown in fig4.

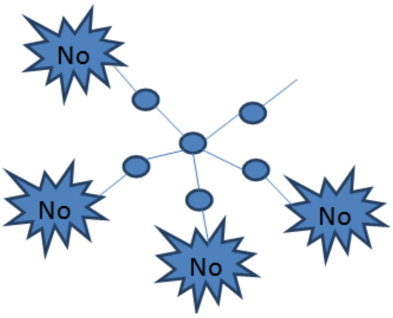


Fig 3: Different nodes with non overlapping group

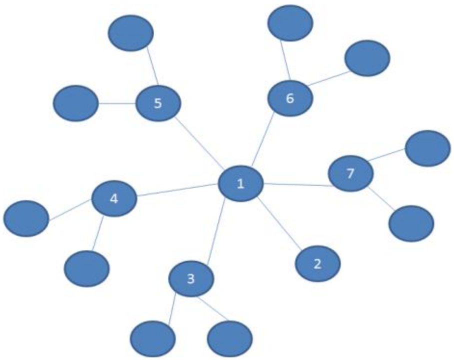


Fig 4: Various Nodes with its associated nodes who share the loads among them

ii. **SSL_LB Scheme algorithm:**

Some unexpected load to nodes of peer to peer system degrades the heterogeneous system. To overcome of this problem we can connect load balancer with secured socket layer, which facilitates efficient way to distribute work. This model save and serve the equal loads to the nodes that algorithm name is SSL_LB and RSA algorithm[6]. This algorithm finds out response time of each request from client to server. So, SSL based algorithm for load balancing give better performance in four measures

1. Security
2. Throughput
3. Coverage
4. Latency

The existing models consist of Shortest Expected Delay (SED) and Never Queue Policy (NQ). In NQ policy all the nodes are with empty queue having load. If any of nodes is non empty then SED policy is invoked. Some of the limitation of existing system is schedule decisions are predetermined. And also server allotting process is very low. All these existing models for secured load balancing have same objectives and pros and cons. Some advantages over these models.

- High speed performance.
- High availability and error recovery.
- Increase in extendibility.

iii. Secured load balancing with skew and churn:

In this model many researchers found secured load balancing technique for heavily loaded and high rated node that is skew and churn resp. In this process for load balancing arbitrary locations of nodes are choose along with their ID. By using k-choice algorithm for load balancing this skew and churn earn high security. This algorithm chooses the targeted node along with their capacity of workload[9]. All servers are connected with its associate node. Each node have its own workload and capacity. Distribution of such type of load is shown in fig5.

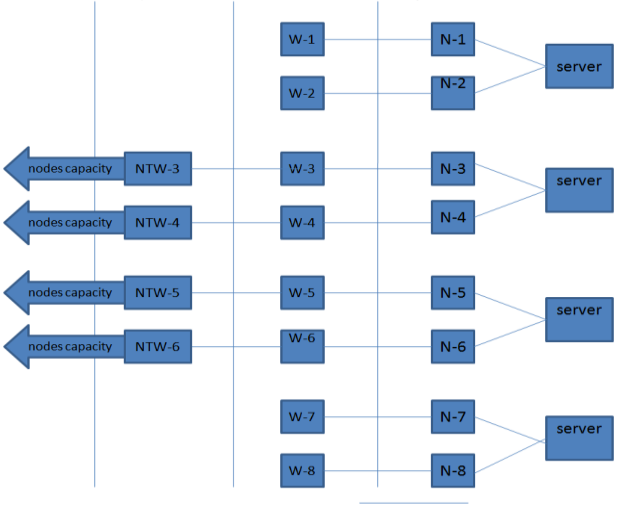


Fig 5: Different servers with its associated nodes and its workload and capacity

Many existing models had achieved the load balancing which means that distribution of load to all nodes is same. In the wide area of cooperative storage for node all the load can be balanced even if we removed virtual servers to longer time[4]. Many existing model refers differs algorithm to reduce load to overloaded node as: one too one, one too many and many to many using virtual server in heterogeneous networks. Some models keep such ideas of algorithms that only increase the complexity and error recovery techniques are degraded. In some papers models were built in tree like structure as k-ary tree. For a simple and inefficient load balance within nodes in peer to peer systems many existing models used some techniques are as follows:-

1) **Address – space balancing: -** Each of node's loads has allocated their own address space but by using distributed hash table address space is not partitioned. Some machines get larger portion then assigned so first of all we have to manage address space balancing[3]. For solution this problem we can use virtual nodes in which each original node imagine to many different nodes and which independently participating in DHT. But by using nodes there are some drawbacks. As we require more storage and large network bandwidth it achieves by original node will check the virtual nodes and if necessary migration of load in the system takes place[7].
2) **Data balancing: -** As we are adjusting load within the nodes, storage database should be maintained. As we know hash table record of item and data. It also keeps records of new join data, exit of data, updated data.

III. SYSTEM DESIGN

In this paper we have implements security load balancing algorithm by java sockets. In this process we have to maintain 4 different modules:

A. Modules:

1) Login Module:

As its name implies client should make login registration for authorization process. If authorization is correct than that user is allowed for further process, else we can send error message to client.

2) Client Module:

Client is first user whose request initializes the whole system.

3) Load Balancer Server Module:

1.**Clients identification:-** Client first of all send the request to server, after checking its authentication , server can proceed process.

2. **Distributing load balancing:** - By checking each nodes load balancer distribute load according to their functionality.

4) Sub-Server Module:

Sub server modules retrieve data from LB server. Sub server performs Data retrieval for each client one by one and makes proportion among them

B. System Architecture:

In this paper we design architecture for load balancing in 2 peers. Each peer has its own memory storage area. These peers have its own clients and virtual servers and the load between such nodes are balanced by load balancer. System architecture of load balancer is shown in fig6.

C. Proposed Algorithm:-

STEP 1: Execute the clients request and response.
STEP 2: keep up the list of files in various file types.
STEP 3: Dispatch data according to client Permission and assign server for each client.
STEP 4: Each request by client the threshold value of node keeps on changing.
STEP 5: Load balancing server responses client to assign server and read response time and vice versa.
STEP 6: According to response time load balancing server progresses client request to suitable servers.
STEP 7: Client permission will be progressed to the next server if the current server overflows with client's request.
STEP 8: Client request files will be adapted from Load balancing server.
STEP 9: Load balancing server maintains a queue for each request, response, and Assigning server for each client.

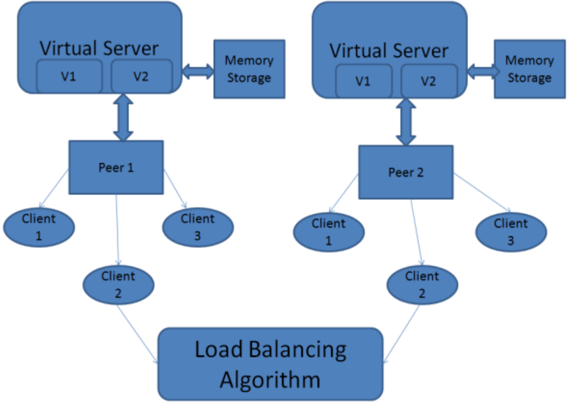


Fig 6:- System Architecture of Load Balancer.

D. Data Flow Diagram:-

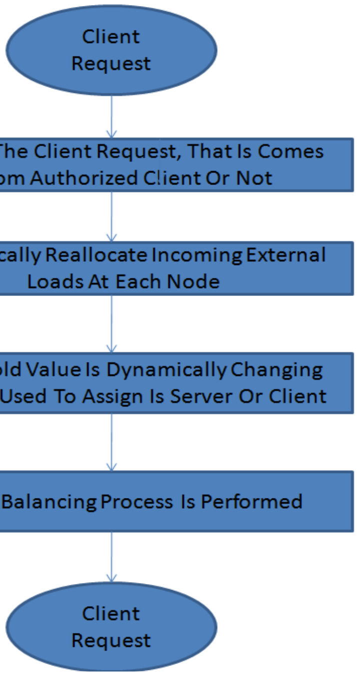

Client Request
Check The Client Request, That Is Comes From Authorized Client Or Not
Dynamically Reallocate Incoming External Loads At Each Node
Threshold Value Is Dynamically Changing Value Is Used To Assign Is Server Or Client
Load Balancing Process Is Performed
Client Request

Fig 7: - Data flow diagram of proposed system

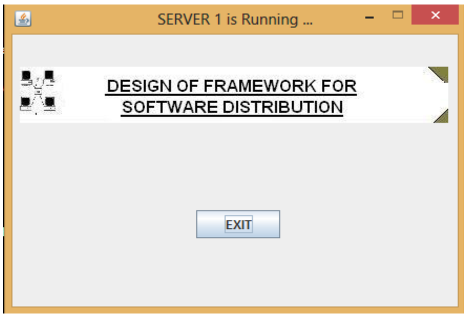
VI. PERFORMANCE ANALYSIS

A) Data Set: -

Variable Used	**Explanation**	**Points**
N	**Node no.**	**2^14**
- **P**	**Mean of systems Potential**	
P	**Items Potential**	
VS	**No. Of virtual server per potential in network**	**2logN**

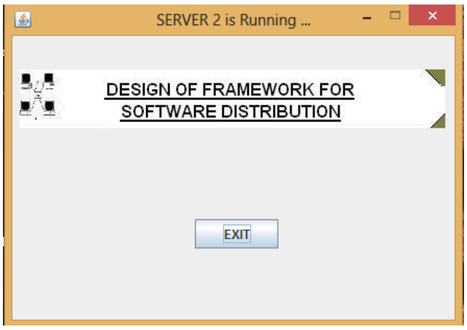
V. RESULT

<u>SUBSERVER 1</u>:-


SERVER 1 is Running ...

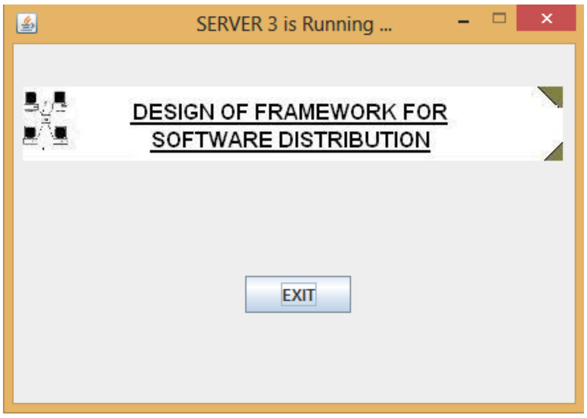
DESIGN OF FRAMEWORK FOR SOFTWARE DISTRIBUTION

<u>SUBSERVER 2</u>

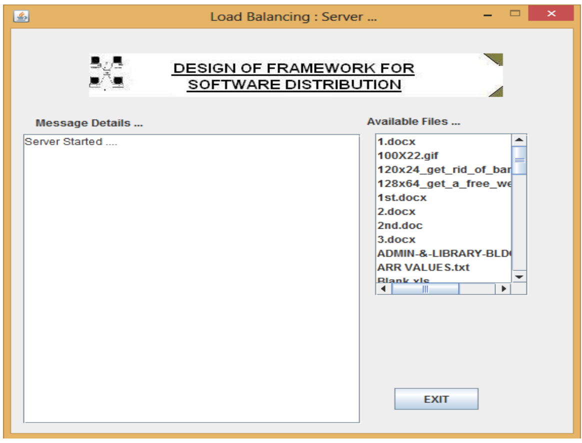

SERVER 2 is Running ...

DESIGN OF FRAMEWORK FOR SOFTWARE DISTRIBUTION

SUBSERVER 3:-



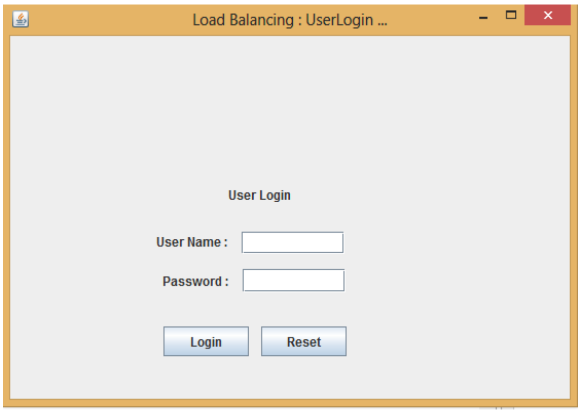
MAIN SERVER:



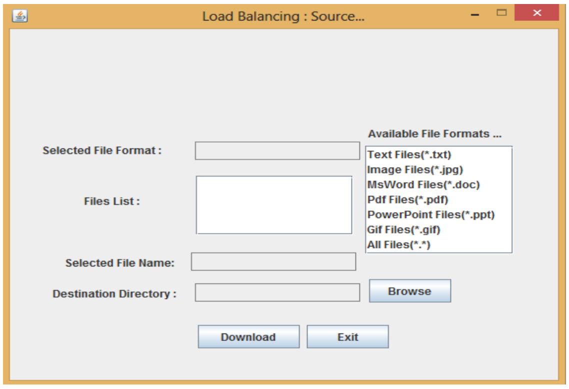
```
Server Started ....
```

```
1.docx
100X22.gif
120x24_get_rid_of_bar
128x64_get_a_free_we
1st.docx
2.docx
2nd.doc
3.docx
ADMIN-&-LIBRARY-BLD
ARR VALUES.txt
Blank.xls
```

LOGIN:-

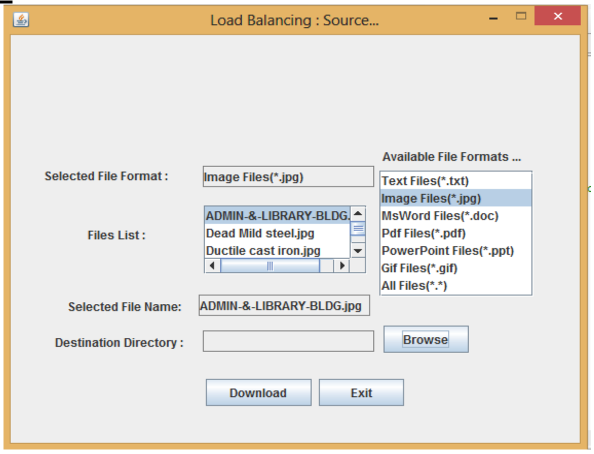


AVAILABLE FILES FORMAT:-



```
Text Files(*.txt)
Image Files(*.jpg)
MsWord Files(*.doc)
Pdf Files(*.pdf)
PowerPoint Files(*.ppt)
Gif Files(*.gif)
All Files(*.*)
```

FILE FORMATE SELECTION



Load Balancing : Source...

Selected File Format : Image Files(*.jpg)

Available File Formats ...

- Text Files(*.txt)
- Image Files(*.jpg)
- MsWord Files(*.doc)
- Pdf Files(*.pdf)
- PowerPoint Files(*.ppt)
- Gif Files(*.gif)
- All Files(*.*)

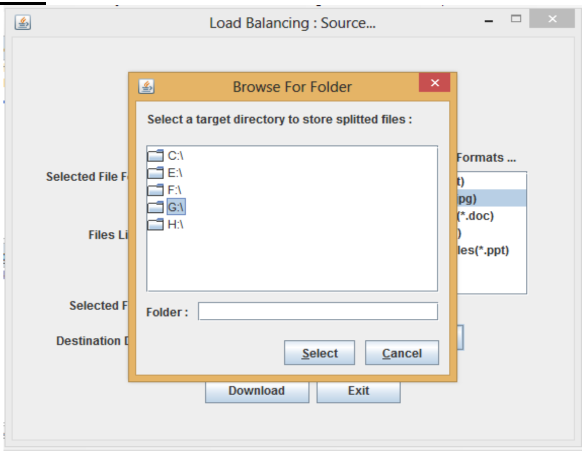
Files List :

- ADMIN-&-LIBRARY-BLDG.
- Dead Mild steel.jpg
- Ductile cast iron.jpg

Selected File Name: ADMIN-&-LIBRARY-BLDG.jpg

Destination Directory :

FILE SELECTION BROWSING :-



Load Balancing : Source...

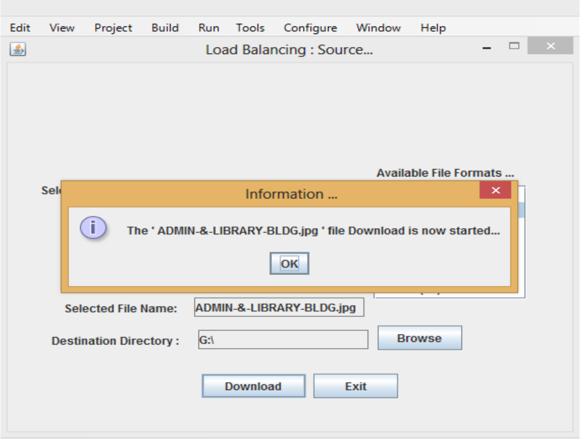
Browse For Folder

Select a target directory to store splitted files :

- C:\
- E:\
- F:\
- G:\
- H:\

Folder :

FILE DOWNLOADING :-



Load Balancing : Source...

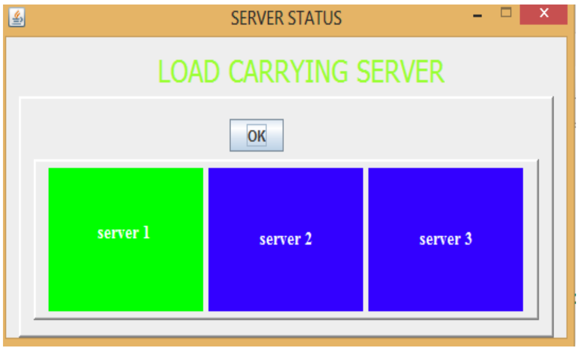
Information ...

The ' ADMIN-&-LIBRARY-BLDG.jpg ' file Download is now started...

Available File Formats ...

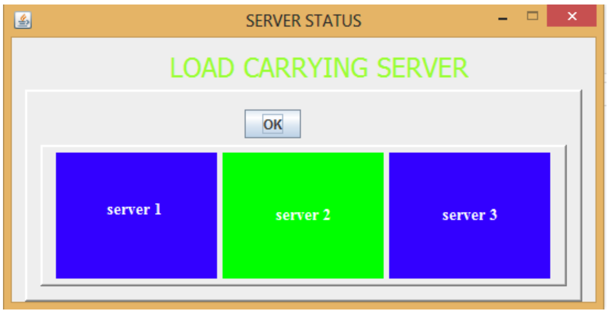
Selected File Name: ADMIN-&-LIBRARY-BLDG.jpg

Destination Directory : G:\

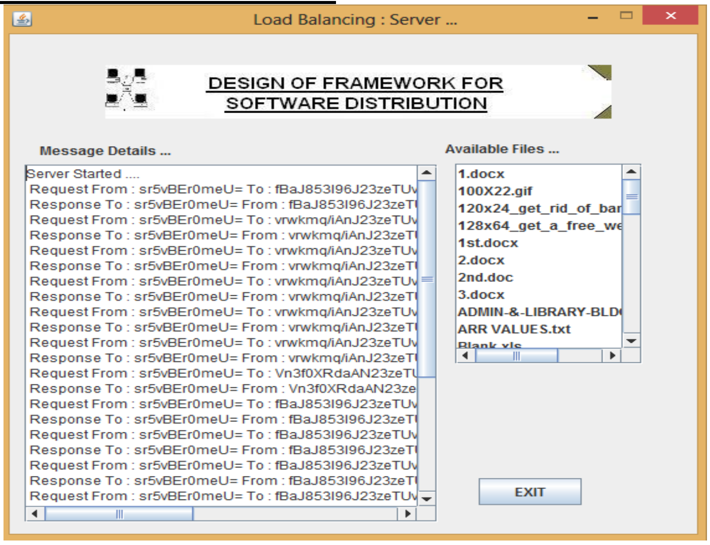


SERVER STATUS

LOAD CARRYING SERVER



DECRYPTED INFORMATION ON MAIN SERVER:-



```
Server Started ....
 Request From : sr5vBEr0meU= To : fBaJ853I96J23zeTUv
 Response To : sr5vBEr0meU= From : fBaJ853I96J23zeTU
 Request From : sr5vBEr0meU= To : vrwkmq/iAnJ23zeTUv
 Response To : sr5vBEr0meU= From : vrwkmq/iAnJ23zeTU
 Request From : sr5vBEr0meU= To : vrwkmq/iAnJ23zeTUv
 Response To : sr5vBEr0meU= From : vrwkmq/iAnJ23zeTU
 Request From : sr5vBEr0meU= To : vrwkmq/iAnJ23zeTUv
 Response To : sr5vBEr0meU= From : vrwkmq/iAnJ23zeTU
 Request From : sr5vBEr0meU= To : vrwkmq/iAnJ23zeTUv
 Response To : sr5vBEr0meU= From : vrwkmq/iAnJ23zeTU
 Request From : sr5vBEr0meU= To : vrwkmq/iAnJ23zeTUv
 Response To : sr5vBEr0meU= From : vrwkmq/iAnJ23zeTU
 Request From : sr5vBEr0meU= To : Vn3f0XRdaAN23zeTU
 Response To : sr5vBEr0meU= From : Vn3f0XRdaAN23ze
 Request From : sr5vBEr0meU= To : fBaJ853I96J23zeTUv
 Response To : sr5vBEr0meU= From : fBaJ853I96J23zeTU
 Request From : sr5vBEr0meU= To : fBaJ853I96J23zeTUv
 Response To : sr5vBEr0meU= From : fBaJ853I96J23zeTU
 Request From : sr5vBEr0meU= To : fBaJ853I96J23zeTUv
 Response To : sr5vBEr0meU= From : fBaJ853I96J23zeTU
 Request From : sr5vBEr0meU= To : fBaJ853I96J23zeTUv
```

```
1.docx
100X22.gif
120x24_get_rid_of_bar
128x64_get_a_free_we
1st.docx
2.docx
2nd.doc
3.docx
ADMIN-&-LIBRARY-BLD
ARR VALUES.txt
Blank.xls
```

VI. CONCLUSION AND FUTURE WORK

This paper examines different existing models for load balancing in peer to peer system. We have also studied the mechanism of SLBA algorithm along with SALT algorithm using virtual servers. This algorithm takes ID manipulation of node which assign load to their fundamental capacity. We have also examined algorithm and different security troubles raised during assign load to different node. We also examine and find result of different parameters which give error free output and load are equally distributed among nodes.

Our future work considers the implementation and evaluation of the proposed load balancing algorithm.

REFERENCES

[1] M. M. Nalgirkar, N. D. Kale “Ample Range Survey On Secured Load Balancing Algorithm In Peer To Peer Network”, Vol-l4, april-2014,International Journal of Advanced Research in Computer Science And Software Engineering

[2] Wei MI, Chunhong ZHANG, Xiaofenf QIU,“SLBA: A Security Load-Balancing Algorithm for Structured P2P,”in Journal of computational Information System 8: 7 (2012) 2760.

[3] J. Byers, J. Considine,”Simple load balancing for distributed hash tables”, LNCS,2003,PP.80-87.

[4] P. Godfrey and I. Stoica, ”Heterogeneity and load balance in distributed hash tables”, INFOCOM 2005, 2005, vol. 1, pp. 595-606..

[5] I. Stoica, R. Morris, ”Chord: A scalable peer-to-peer lookup service for Internet applications”,Proc. of the 2001 Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communications, 2001, pp. 149-160.

[6] A. R. Karthik, K. Lakshminarayanan, ”Load balancing in structured P2P systems”, LNCS, 2003, pp. 68-79.

[7] I. Sit, R. Morris. ”Security considerations for peer-to-peer distribution hash tables”, Future Directionsin Distributed Computing, Springer-Verlag, pp. 103-107, 2003.

[8] S. Ratnasamy, P. Francis, ”A scalable content-addressable network”, Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications, 2001, pp. 161-172.

[9] B. Godfrey, K. Lakshminarayanan, ”Load balancing in dynamic structured P2P systems”, INFOCOM 2004, 2004, vol. 4, pp. 2253-2262.

[10] J. Ledlie, M. Seltzer, ”Distributed, secure load balancing with skew, heterogeneity and churn”, INFOCOM, 2005, vol. 2, pp. 1419-1430.

[11] Jochem van Vroonhoven, ”Peer to Peer Security”, 4th Twente Student Conference on IT, Enschede, 2006.