

Report Title

BUILDING A DISTRIBUTED KEY VALUE STORE-

BIG DATA 2017 - CLASS PROJECT

|  |  |  |  |
| --- | --- | --- | --- |
| SNo | Name | USN | Class/Section |
| 1 | JAMPALA SREECHANDANA | 01FB15ECS132 | 5th/c |
| 2 | KEERTHANA NAGARAJ | 01FB15ECS147 | 5th/c |
| 3 | SHRAVANTHI.R | 01FB15ECS219 | 5th/D |
| 4 | ROHINI.D.V | 01FB15ECS242 | 5th/E |

Introduction

A distributed data store is a computer network where information is stored on more than one node, often in a replicated fashion. It is usually specifically used to refer to either a distributed database where users store information on a number of nodes, or a computer network in which users store information on a number of peer network nodes.

## Related work

<https://zookeeper.apache.org/>

Studied the main pages to understand the server client connections.

## ALGORITHM/DESIGN

1.Creation of a Client and Server java files using socket programming.

2. Start ZooKeeper and creation of Master Znode

3. Check for the presence of Master using ZooKeeper

SERVER OPERATION

· Clients will send requests to the server

· Server will determine request type – put, get

· Server will determine if it can process the request or the request has to be serviced by other servers

· For self-served requests – it will process the request and send back status of response

SERVER REPLICATION

· Based on the server name a hash code function assigns a random number.

· This random number is used to assign the last 8 bytes of the IP Address for the server.

Example: If the hash code returns a value 1234. Then required value = (1234) %255 Required value = 214

So, IP Address for the server will be 127.0.0.214 Its replica would be hash code return value of hashcode(servername+r).

HANDLING SERVER FAILURE

· Client tries connecting to server with key.

· On server failure, connects to master to get new list of keysserver mapping.

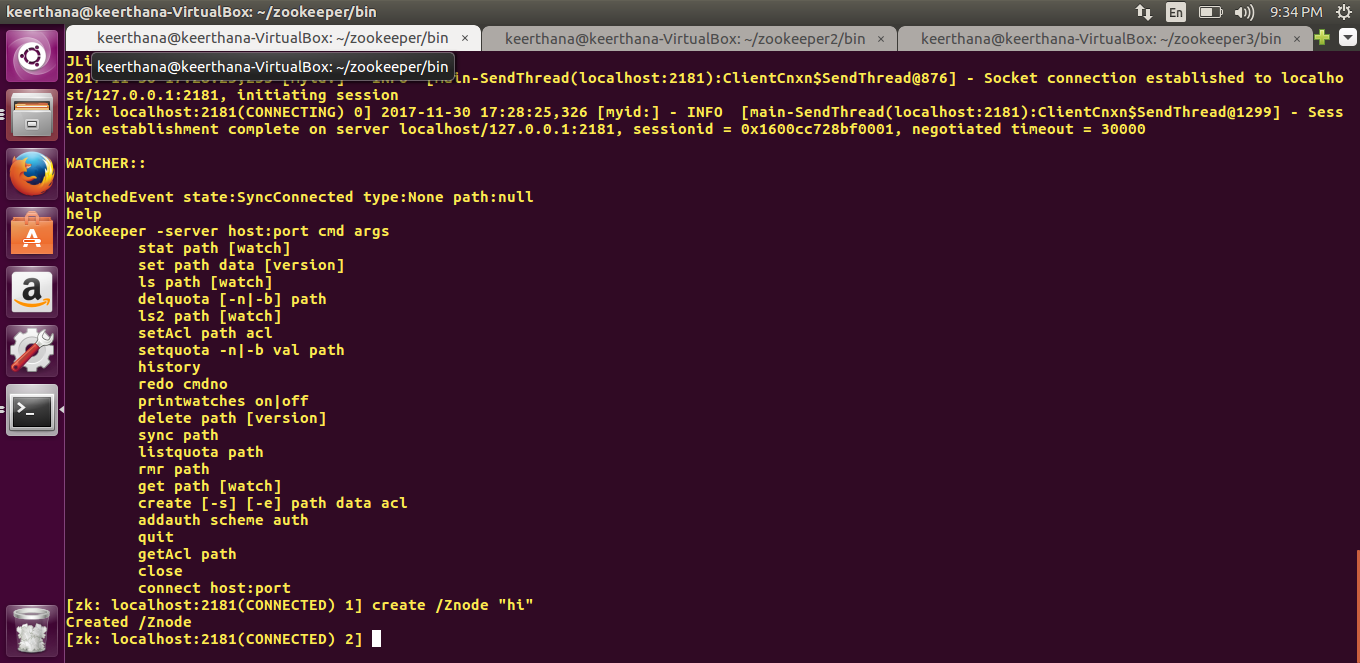
· Talks to the replica to retrieve data

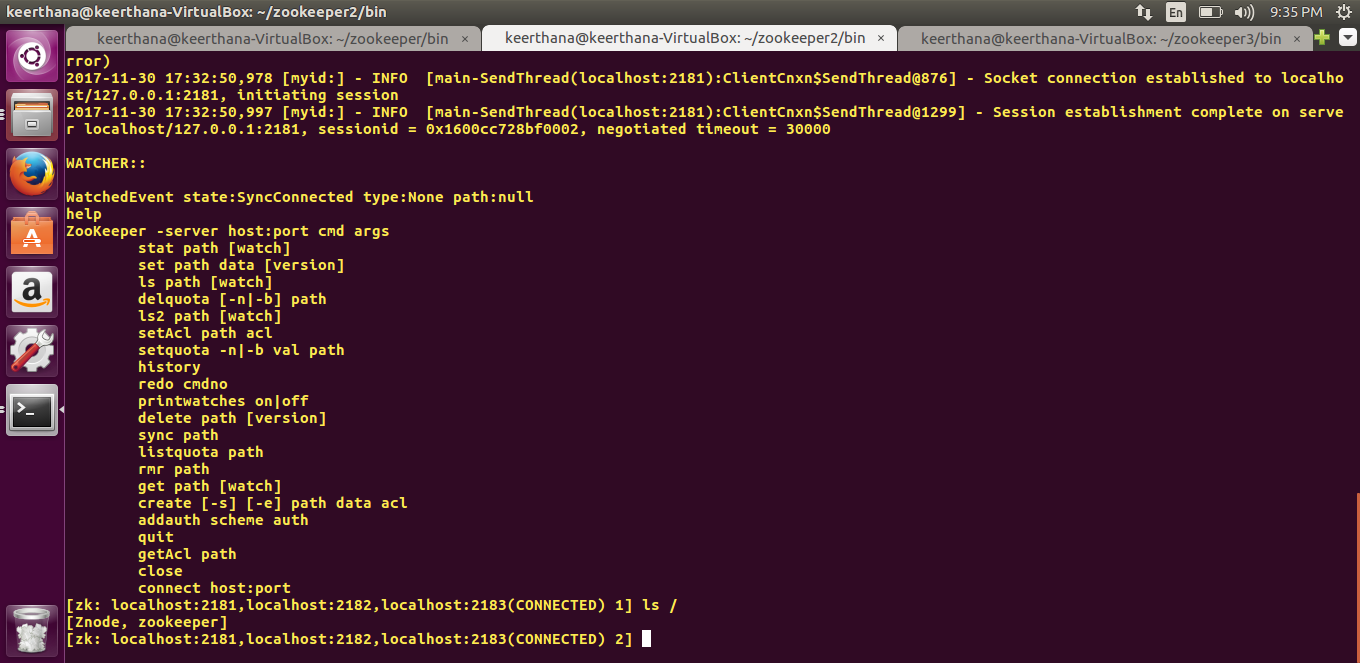
## EXPERIMENTAL RESULTS

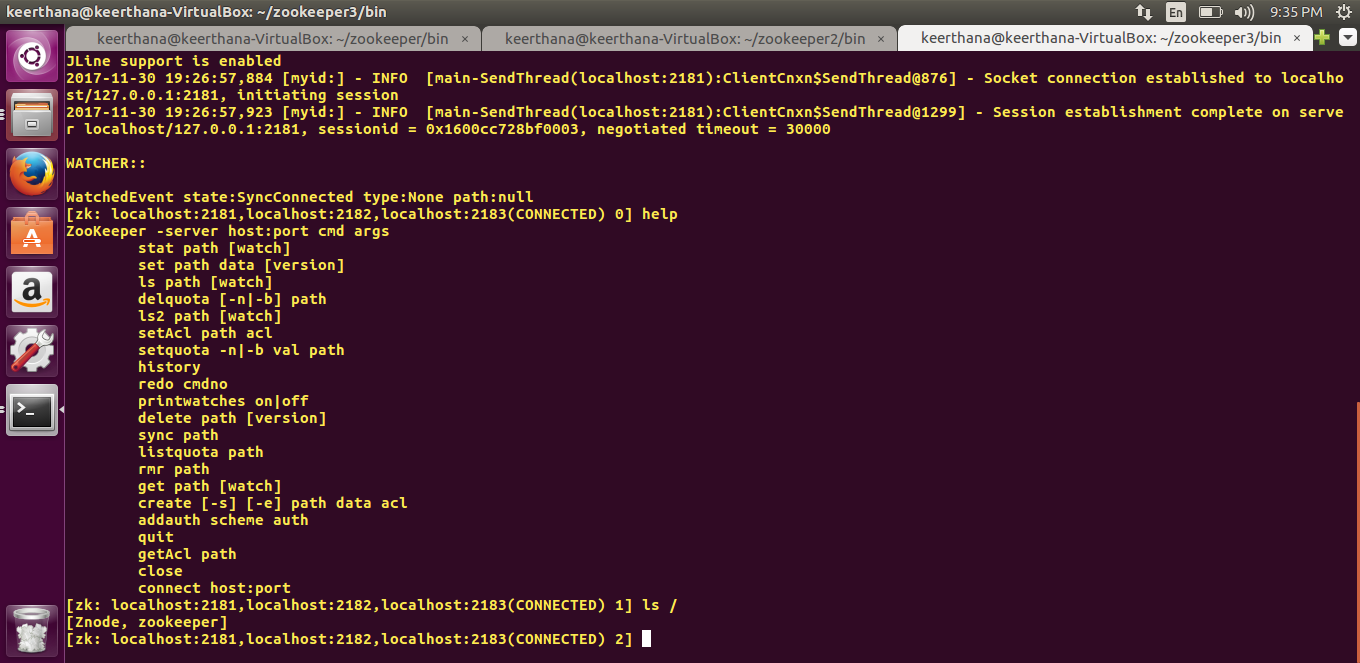
· Successfully established connection between client and server.

· Successful querying of keys by the client from various servers with distributed key value pairs.

· Server failure handled, its contents replicated in a replica-server and client retrieval from replica-server.







## FUTURE ENHANCEMENTS

Handling additional servers, more than three

## REFERENCES

<https://askubuntu.com/questions/>

<https://tutorialpoint.com/zookeeper>

<https://zookeeper.apache.org/hadoop/zookeeper>

<https://myjeeva.com/zookeeper/clustering/setup.html>

<https://javatpoint.com/socket-programming>