© 2014 IJIRT | Volume 1 Issue 5 | ISSN : 2349-6002

JAVA NETWORK PROGRAMMING USING SOCKET

Gourav Rathor, Devankv. Rathore

Abstract- **This paper depicts about Network programming utilizing java. The Network writing computer programs is like attachment programming or Client-Server programming. Where Socket writing computer programs is imperative to see how web based interprocess correspondence work. In this we depict about distinctive sorts of attachment utilized as a part of interprocess impart on. System programming fundamentally utilizes the Client Server model. In Client-Server programming there are two separate projects or procedure, one which launchs correspondence called Client methodology and other who holding up for correspondence to begin is called Server process. In this paper we additionally concentrate on the Secure Socket Layer required for security reason. At last correlation between the Networks programming utilizing C dialect and Network Programming utilizing Java is indicated.**

I. INTRODUCTION

In today's reality Internet is utilized by every and everyone. The web is about interfacing machine together and correspondence. This is the place Network Programming comes. System programming permits Inter process Communication. It implies it includes composing machine programs that speak with other program over a machine system. System project can do heaps of work. A straightforward system Program can get data from numerous machines spotted everywhere throughout the world. It can correspond with a large number of individuals around the globe.

System programming uses Client-Server demonstrate, so system writing computer programs is likewise Client-Server Programming. Where one project begin the correspondence called customer process or system and other who is sitting tight for correspondence to begin called the server process or system. In the most straightforward case, Client project sends the appeal to the server. Server sends the reaction. Customer then gets the information from the server and showcases it. Complex customers channel and rearrange information, over and again get changed information, send information to other individuals , and permits ongoing talking, multiplayer gaming. Complex servers regularly do a ton of preparing on the information before noting the inquiry. System programming makes utilization of attachment for interprocess correspondence. Where attachment go about as the end point of the inter process correspondence. Here attachments can likewise be termed as system attachment or Internet attachment since correspondence between machines is focused around Internet convention. So Network writing computer programs is like wise Socket Programming.

From last few years system programming has quit being regional unit of the pro and got to be a piece of each engineers work midsection. It was begun to being utilized as a part of all application. So there is have to make it more straightforward, So Java was begun to being utilized for system programming in light of the focal points Java gave.

Java was the first programming dialect outlined in light of systems administration. Java was initially intended for restrictive HQ telecom companies instead of the Internet, however it’s generally had the system preeminent as a primary concern.

Java gives answers for various issues. An alternate playing point of java is that it gives security. Greatest playing point of java is that it makes composing the projects less complex. It is far easier to compose system program in java than in any dialect. The system program written in java would be less demanding to comprehend and more averse to have lapse than project written in other dialect. In Java piece of program that arrangements with system is constantly short and basic. In short it is simple for java application to send and get information furthermore to impart over the web.

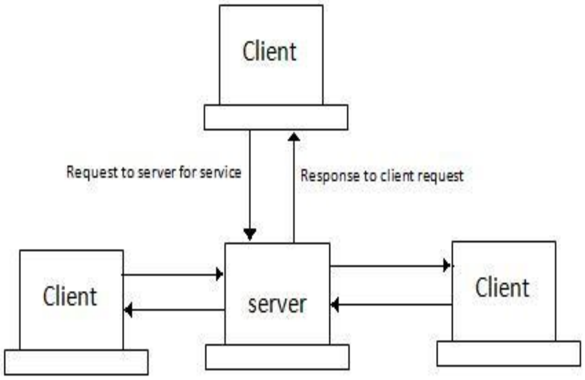
Java incorporates classes that help system project correspond with specific sorts of servers and procedure distinctive sorts of information yet not all the servers and information. So Java permits you to compose convention handlers to speak with distinctive server and methodology the information. The most energizing gimmick of Java is it contains a simple to utilize and cross stage model for correspondence which make system programming straightforward rapidly.

II. RELATED WORK

In this Section the center is fundamentally on the Client-server model, Socket programming and the protected attachment Layer (Ssl).in Client server model segment distinctive sorts of important programming models are portrayed. In attachment programming we portray about the attachments, capacity calls, sorts of attachment and so forth. Secure attachment layer is essentially needed for security motivation behind the projects.

A. Client Server Model

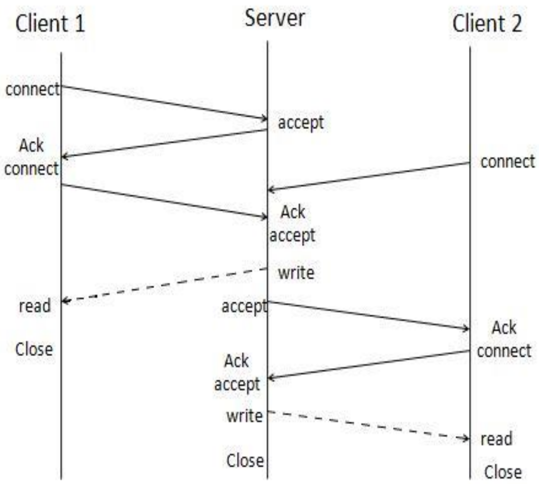
In Client Server Program there are two principal programming models. They are Iterative Server model and Concurrent Server Model.



Network programming uses Client-Server Model. Client program or process initiates the communication and servers are program or process who waits for communication to start. Some time program may be both a client and server. Here Client sends the request to which the server sends the response. Typically we have single server multiple client's model. The server does not need to know anything about client even that it exists or not. The client should always know something about the server atleast where it is located. The IP and Port number of the server is generally well known. So the client knows where to send the request. In Contrast the port number on client side is generally allocated automatically by the Kernel.

I) Iterative Server Model

In Iterative Server Model Only one solicitation is handled at once. In this audience and server segments of the application exist together and run as a feature of the same errand. The server application holds the attachment until all application handling has finished. In this customer needs to hold up until server does not react. It is Easy to manufacture and it is generally utilized when appeal can be finished in a little time. The issue with this customer server model it causes unnecessary postponement.

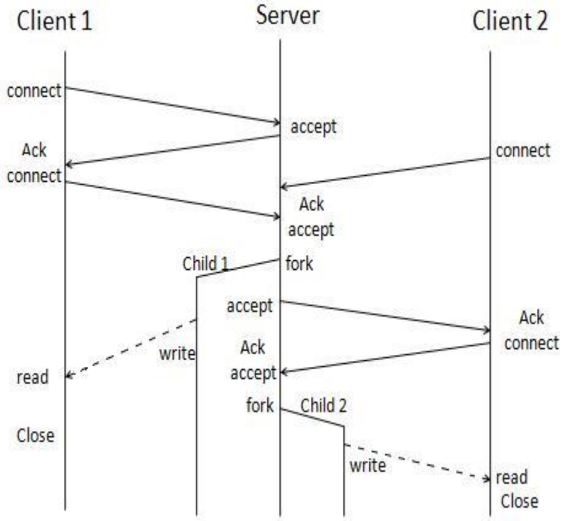


An ordinary Iterative server utilizing attachment is indicated within Figure.3.in this first attachment is made. At that point tie the attachment to well-known port. At the point when the solicitation from the customer has been gotten the server sends the answer to the customer. No new ask for will be handled until the past solicitation is prepared. After solicitation is prepared passageway and server acknowledge new ask for and takes after the same strategy.

```
Algorithm Iterative Server

begin
      Create Socket

      bind to well Know port
      while(1){

              read request from client send
              reply to client exit
              }
end
```

II) **Concurrent Server Model**

In Concurrent Server Model numerous appeal are transformed at same time. In this the server part is to listen for administration demand from distinctive host or customers. When the solicitation has been gotten the servers fork the kid procedure to handle it and server backpedals to listening to other appeal. In this new ask for can be transformed while other ask for even now being served. It gives preferred execution over Iterative server model. The disservice of this model is it is hard to outline and construct.



A common Concurrent server utilizing attachment is indicated within Figure.5.in which first attachment is made. At that point tie the attachment to well-known port. At the point when the solicitation from the customer has been gotten the system association is setup. The server then forks a youngster procedure to serve the appeal from customer and it about-faces to listen to other customer demand. Here new ask for is acknowledged regardless of the fact that past one is even now staying to be served.

Algorithm Concurrent Server

```
begin

       create socket

       bind to well know port
       while (1){

              read request from client
              if(fork()==0){//child

                     Serves request from client

                     exit
                     }

              else{

                            //parent

                     }

       }

End
```

B. Socket Programming

System programming makes utilization of attachment for Interprocess Communication. Because of which Network writing computer programs is additionally termed as attachment programming. In Socket programming utilizing Java, BSD style Socket to Interface with TCP/IP administrations is utilized. BSD Socket Interface gives offices to Interprocess Communication. BSD Socket Interface Supports diverse area, the UNIX Domain, the

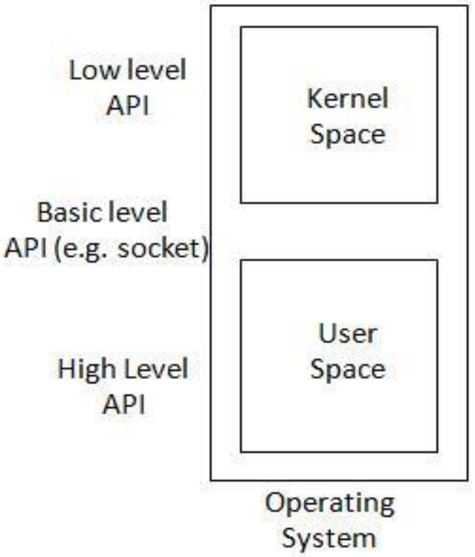
Web Domain and the NS Domain. Java Basically backs the Internet Domain to keep up cross stage. In Internet Domain, the BSD Socket Interface is based on the highest point of either TCP/IP or UDP/IP or the crude Socket. Attachment Programming is paramount to see how web based interprocess correspondence work however not at the level project created yet at a larger amount that is assembled to situated of Socket Programs. Before going in insight about Socket we should concentrate on the

Correspondence API.

I) **Communication API**

Correspondence APIs empower access to administrations for nearby and remote correspondence. They are available in registering framework at distinctive level. There are essentially three levels High level Communication Apis, Low Level Communication Apis and Basic Level Communication Apis.

Abnormal state Communication Apis are regularly show in the User Space which gives programming dialect methodology to middleware correspondence administration. Low level Apis are available in part space. The Basic level Communication Apis are normally spotted between the client space and Kernel space. It generally permits application projects to increase administrations for Interprocess correspondence, system correspondence and gadget -correspondence. At this Basic Level Communication Sockets are available which is utilized for Interprocess Communication i.e for Network Programming.



II) **Sockets**

Socket is an Interface between the application and the system. Where Application make attachment and attachment sort manages style of correspondence i.e. connectionless or association arranged. Attachment essentially goes about as endpoint of Interprocess correspondence. The interprocess correspondence over Internet is focused around forms of attachment. Attachments can likewise be termed as system attachment or Internet attachment since correspondence between machines is focused around Internet convention. Socket are a method for utilizing IP to convey between machines. Attachment is one significant gimmick that permits Java to interoperate with legacy framework by simply conversing with existing server utilizing predefined convention. Attachment additionally has some transparency that permits portion to redirect yield of one procedure to the information of an alternate machine.

Socket is an Interface between the application and the system. Where Application make attachment and attachment sort directs style of correspondence i.e. connectionless or association

i) Managing Connection:-

In this we have socket(), bind(), listen(), connect() and accept(). Where socket() call is used to create a Socket. An application opens a socket by using the socket() call. bind() call is used to bind a socket to local IP address and port. listen() call is used in passive open .it is used by server to announce its objective of receiving incoming connection request. connect() call is used to start connection to another socket. accept() call is used to accept a new connection.

ii) Sending and Receiving Data:-

In this we have send(), sendto(), recv(), recvfrom() and etc. Where send() call sends data on a connected socket. sendto() call is used to send the datagram to another UDP socket. recv() call is used to receive message from socket. recvfrom() is used to read a datagram from a UDP socket.

iii) Managing Endpoint Characteristics:-

In this we have Close() function which is basically used to close the socket i.e. shutdown the connection.

Sockets are divided into different types. They are distinguished between TCP Client Socket, TCP Server Socket and UDP Socket.

i) TCP Socket:-

TCP Socket is that socket which is used for more reliable connection. They perform more reliable and in-order delivery of the packets. They are used when there is Connection oriented Communication.TCP socket is also called Stream Socket. TCP Socket supports out of bound data transmission. In TCP Socket there is TCP Client Socket and TCP Server Socket.

a) Client Socket

The Client Program or process basically uses the TCP Client Socket. Client Socket are also just called Sockets. To implement the Client Socket the Client program makes use of java.net.Socket

class present in java.net package. The java.net.Socket class is used for doing client side TCP operations. Client Socket act as an endpoint for communication between two machine. The Socket class contains constructor that is used to create stream socket that connects to specific port number associated with exactly one host.TCP Client Socket encapsulates a java.io.InputStream and java.io.OutputStream.

Client program written using Client Socket will have following life cycle:-

1) Creates a stream socket using constructor Socket(String host, int port).

2) Sockets try to connect to remote host.

3) Then, Input stream and output stream are used by client and server to send data to each other. Both Client and server agree upon handshaking before sending data.

4) When communication is over or data transfer is complete one or both side close the connection.

b) Server Socket:-

The Server Program or process basically uses the TCP Server Socket. To implement server sockets java.net.ServerSocket class is used which is present in the java.net package. The java.net.ServerSocket class is used for doing server side TCP operations. It waits for request to come over the network and then perform operation based on request. Once ServerSocket has setup connection the server uses Socket object to send data to client. The ServerSocket Class contains Constructor that is used to create new ServerSocket Object on a particular local port and also contains method like accept() to listen for connection on a specified port.

Server program written using Server Socket will have following life cycle:-

1) Creates a ServerSocket on a particular port using constructor ServerSocket().

2) Listen for connection to be made to this ServerSocket by using accept() method. This method waits till client connects to server and then returns the Socket object.

3) Then, Input stream and output stream are used by client and server to send data to each other. Sever hear the client using input stream. Server talk to client using output stream.

4) Both Client and server agree upon handshaking before sending data.

5) When communication is over or data transfer is complete one or both side close the connection.

6) The Server then returns back to second step and waits for next connection.

ii) UDP Socket:-

UDP socket uses User Datagram Protocol (UDP) alternative protocol for sending data. UDP provides connectionless communication. The connection here would be unreliable. It does not have starting handshaking phase and here data received would not be in order or lost. Doing socket programming using UDP no streams are attached to socket.UDP socket is also called as DatagramSocket. Datagram Socket is opened to send and receive DatagramPacket.

To implement the Datagram Socket the program make use of java.net.DatagramSocket Class which is present in java.net package.DatagramSocket class contains Constructor that is used to construct a datagram socket and bind it to the specified port on the local host machine. It also contains method like receive() and send() to receive and send the datagram packets. In this Socket used by client and the server are almost same only thing they differ in whether they use anonymous or well-known port.So there are no special socket class for server.

Client program written using Datagram Socket will have following life cycle:-

1) Creates a Datagram socket using constructor DatagramSocket().

2) Translate hostname to IP address.

3) Create the Datagram Packet using DatagramPacket (Data,Data.length,IP,Port)

4) Send datagram Packet to server using send().

5) Read datagram Packet from Server using receive().

6) When communication is over or data transfer is complete close the connection.

Server program written using Datagram Socket will have following life cycle:-

1) Creates a DatagramSocket on a particular port using DatagramSocket(int port).

2) Read Datagram Packet from client using receive().

3) Get IP address and port no. of client.

4) Create the Datagram Packet using DatagramPacket (Data,Data.length,IP,Port)

5) Send datagram Packet to client using send().

6) The Server then returns back to second step and waits for next connection.

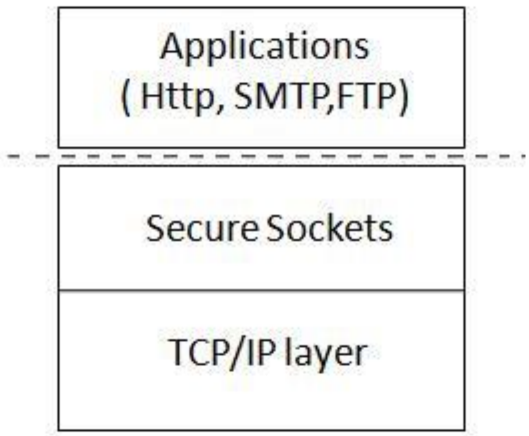
C. Secure Sockets

Security is important on Internet and people want authentication, integration and privacy. Secure Socket layer(SSL) protocol basically helps in achieving this security goals. SSL was developed by Netscape as security measure for web server and web browser. At present SSL is largely used in intranets and also in many public internets. SSL has become de facto standard for providing secure e-commerce transaction over the web.

SSL allows web browser to authenticate the web server. SSL requires web server to have digital certificate on it for SSL connection to be made.SSL is mostly used as Secure transport below HTTP. Secure hypertext transfer protocol i.e. HTTPs is a http who is using the SSL.SSL basically sit on top of the TCP i.e. it lays between TCP and other upper layer can be seen in figure. Developer takes advantage of this by replacing all TCP socket call with the new SSL calls.

SSL consist of two phases: Handshake and data transfer. During handshake phase Client and Server use a public key encryption Algorithm to determine secret key parameters. In this phase the communicating parties optionally authenticate each other and then exchange the session key. During Data transfer phase both sides use the Secret key to encrypt the data transmission.

SSL is an Asymmetric protocol. It differentiates between a client and server. The SSL handshake sequence may vary depending on whether the RSA or Diffie Hellman key exchange is used. Mostly SSL session make use of RSA key exchange algorithm with only server authenticated. Here client authentication is optional and is omitted in most cases.



I) Creating Secure Client Socket

For creating Secure Client Socket instead of creating or constructing java.net.Socket object with a constructor ,use javax.net. ssl . SSL Socket Factory to create Client socket using its createSocket() method. SSLSocketFactory is an abstract class that follows the abstract factory design pattern: public abstract class SSLSocketFactory extends SocketFactory

Since SSL Socket Factory is itself abstract you get an instance of it by invoking the static SSLSocketFactory.get Default() method. Once reference to the factory is done, then createSocket() methods to build an SSLSocket. The Socket that create Socket() method return will be javax .net .ssl. SSL Socket a subclass of java.net.socket. Once secure socket is created use it like any other socket through its getInputStream(), getOutputStream() and other methods.

II) Creating Secure Server Socket

For creating Secure Server Socket instead of creating or constructing java.net.ServerSocket object with a constructor, use javax.net.SSLServerSocket to create Server socket. Like SSL Socket all the constructors in this class are protected.Like SSLSocket, instance of SSLServerSocket are created by an abstract factory class javax.net.SSLServerSocketFactory.

Also like SSLSocketFactory an instance of SSLServerSocketFactory is returned by a static

SSLServerSocketFactory.getDefault() method. Like SSLSocketFactory, SSLServerSocketFactory has CreateServerSocket() method that returns instance of SSLServerSocket. The factory that SSLServerSocket Factory.getDefault() returns generally only support server authentication. It does not support encryption. To get encryption server side secure socket requires more initialization and setup.

III) Secure Socket Benefit

Secure Socket Layer Provides the following Benefits:-

1) Authentication: - It ensures that the end systems are the same systems that they say they are.

2) Message privacy: - It ensures that the data exchanged is not viewed by other person other then receiver even if it is intercepted by unauthorized party.

3) Integrity: - It ensures that the data is not modified or tampered over the Internet.

III. DISCUSSION

In this area correlation between system programming in java and system programming in C is demonstrated.

System programming utilizing C had a few impediments because of which Network programming utilizing Java is utilized. In Network programming in C composition the system project were Difficult to compose likewise they were hard to comprehend and they are length. Additionally C is not cross stage and it doesn't give code convenientce. So composing a system project utilizing c as a part of Linux environment won't work in nature's turf. It have less number of library record for system programming contrast with java so composing system project is tiny bit troublesome. It is hard to send and get information for correspondence contrast with java. Likewise it doesn't give so much security. In Programmer written work system project utilizing C ought to have extensive learning of the framework with which they are working with.

Network Programming using java have lots of advantages that why it is preferred over Network programming using C. Writing the network program in java is simpler and network programs are easy to understand and the programs are also small compare to network program written in C. Also Java is cross platform and support code portability. So Java network program written in java in Linux platform will work on windows platform. It has more library file for network programming compare to C so writing network program is simpler. It is easy to send and receive data for communication using java. Also it provides more security compared to C. It has SSL which can provide more security. In Network programming using java network intensive programs like web servers and clients, almost all the code handles data manipulation or the user interface. The part of the program that deals with the network is always short and small. In Network programming in java using socket we have different types of socket class for Client and different socket class for Server. At client side we use socket class and at server side we use ServerSocket class. Another advantage is that java gives option of applet. In applet the code is not allowed any permanent access to the system. So code is executed but no damage to the system. It is also possible for applets to communicate across the internet but they are limited by security restriction.

Network programming Using java we have Java Socket. Java socket has some Advantage and Disadvantage which are as follows:-

Advantages:-

1) Java sockets are flexible and are more powerful.

2) Efficient Socket based programming can be easily implemented for general communication.

3) Java Socket causes low network traffic, if efficiently used.

4) Unlike html forms and CGI scripts that generate and transfer whole web pages for each new request can only send necessary information.

5) Java socket provides a simplified interface to native socket such as BSD and winsock 2.

6) It hides much of the detail involved in traditional socket programming.

Disadvantages:-

1) Security restriction are sometimes over bearing because a Java Applet running in a web browser is only

able to establish Connection to the machine where it came from, and to nowhere else on the network Despite all the useful and helpful feature.

2) Socket based communication allows only to send packets of raw data between applications. Both the client-side and server-side have to provide mechanism to make the data useful in any way.

IV. CONCLUSIONS

This paper on Network programming in java describe in detail about concepts used in network programming in java. It describes about Java network Programming and its application. Network programming is Client server programming, so it describe about different client server model. It also explains about Socket programming, different types of sockets used like TCP or UDP. From security perspective we have SSL in network programming using java. This paper gives details about SSL, Creating secure Socket and its benefit. Finally there is comparison of Network programming using Java and Network programming using C, provided the advantages and disadvantages of both. Network programming using Java have lots of advantage due to which it is preferred.