AccessMe: Telnet-Based Android Application

Shruti Urkude [1], Dhanashri Galgalikar [2], Priyanka Ahir [3]

Dept. of Information Technology
Y. C. C. E, Nagpur, Maharashtra, India

Abstract— In computing, a shared resource is a computer resource made available from one host to other hosts on a computer network. It is a device or piece of information on a computer that can be remotely accessed from another computer, typically via a local area network or an enterprise intranet, transparently as if it were a resource in the local machine. Some examples of shareable resources are computer programs, data, and printers.

AccessMe is a Telnet based application used to control Remote Computers as well as Mobiles. It is a complete solution for all scenarios like:

- Connect from Android Device to other Android devices as well as to Windows, or Linux PCs.
- **Network Related Details:** Port Scanning, Graphical representation of Upload and Download Speed, Latency.
- Simplified File Transfer, Remote Printing, Monitoring-Check, Blocking unwanted access.
- Video Conferencing, Teamwork, Group Chatting and Sharing.

The main objective of this project is to provide convenient way of sharing data, devices and software between mobiles and computers .It is an attempt that enable the user not only to share the resources but also to view network related detail.

I. INTRODUCTION

The remarkable scalability that characterizes peer-to-peer applications, for example content sharing applications, could not be achieved without users cooperation. A key requirement is especially users' contribution with private, often limited, resources such
as files, software, printers, etc. Shared file and printer access require an operating system on the client that supports access to resources on a server, an operating system on the server that supports access to its resources from a client, and an application layer (in the four or five layer TCP/IP reference model) file sharing protocol and transport layer protocol to provide that shared access. Modern operating systems for personal computers include distributed file systems that support file sharing, while hand-held computing devices sometimes require additional software for shared file access.

Telnet is a network protocol used on the Internet or local area networks to provide a bidirectional interactive communication facility using a virtual terminal connection. Telnet is a client-server protocol, based on a reliable connection-oriented transport. Typically, this protocol is used to establish a connection to Transmission Control Protocol (TCP) port number 23, where a Telnet server application is listening. Telnet, however, predates TCP/IP and was originally run over Network Control Program (NCP) protocols. The term 'telnet' may also refer to the software that implements the client part of the protocol. Telnet client applications are available for virtually all computer platforms. Telnet's application-level semantics are captured in its external interface, the Network Virtual Terminal. Its internal interfaces have close ties to TCP transport facilities, an option negotiation scheme, and symmetric treatment of client and server roles. Notice that none of these three internal functions has any bearing on remote-login, the most popular application of Telnet.

The reason behind developing an Android application is that besides enabling communication with the cloud, Android's wireless Application Program Interfaces also enable communication with other devices on the same network, and even which are not on a network, but are physically nearby. The addition of Network Service Discovery (NSD) takes this further by allowing an application to seek out a nearby device running services with which it can communicate
Integrating this functionality into our application will be providing a wide range of features, such as remotely logging into other machine on the same network, getting the shared resources and so on.

II. PROBLEM DEFINITION

The problems that we have considered for this project include:

- Many a time, there is a requirement of resources that are available on a remote machine. The resource can be anything including data, devices such as printer, scanner or a software. Apart from this, we don't have the time to get on to the place where the resources are available.
- On our device, especially on mobile, we find it difficult to get network related details such as Port Scanning, Speed test of Samba server and FTP,

Graphical representation of Upload and Download Speed, Latency.

- People find it difficult in ensuring reliable transmission of data.
- The applications that are available for these usages provide the required features but separately in different applications and do not provide all the features in a single application.
- The last and the most important problem is platform. We found it difficult to share resources on machines with different platforms.

Fig. 1: Cloud showing different information that can be shared by different devices

III. FEATURES

SCREENCASTING

A screencast is a digital recording of computer screen activity, often containing audio narration. The content of computer screen, microphone, video camera and computer audio can be captured with screencast, although it is usually used only for recording computer screen and audio narration. Terms screencasting, screen capturing and screen recording are synonymies, and refer to the process of creating screencast.

Resource Sharing

A user will be able to share any kind of resource such as File, Printer, Scanner, etc.

Text Sharing

Using this feature we can use AccessMe as a "Chatting Application".

Network related Features

Port Scanning, Graphical representation of Upload and Download speed, latency.

IV. SYSTEM ANALYSIS AND DESIGN

In order to provide connection between system, the basic input required by the application is their IP addresses. Also for providing sharing facility, the basic requirement is that the systems must be in a network.

If you are using internet connection in your android application, then it is important to check the internet connectivity before doing any other operations. It is simple to check the internet connectivity status in android. **ConnectivityManager** class given by android sdk gives you ability to:

1. Monitor network connections (Wi-Fi, GPRS, UMTS, etc.).
2. Provides an API that allows applications to query the state of the available networks.

1) 5.1 Connection Establishment:

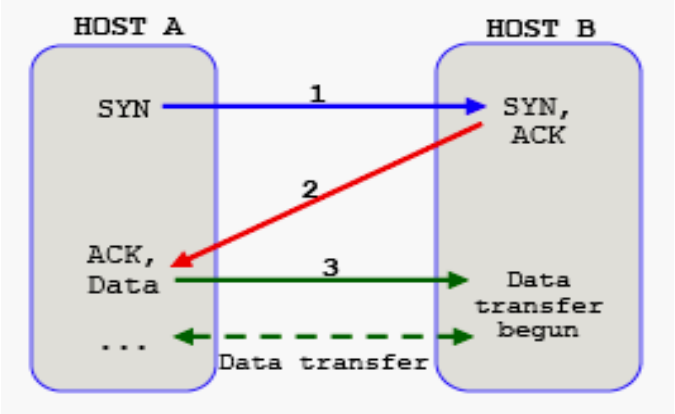

Fig.2 : TCP Three-Way Handshake

Apart from this, we need to provide reliable connection. To establish a connection, TCP uses a three-way handshake. Before a client attempts to connect with a server, the server must first bind to and listen at a port to open it up for connections: this is called a passive open. Once the passive open is established, a client may initiate an active open. To establish a connection, the three-way (or 3-step) handshake occurs:

1. **SYN**: The active open is performed by the client sending a SYN to the server. The client sets the segment's sequence number to a random value.
2. **SYN-ACK**: In response, the server replies with a SYN-ACK. The acknowledgment number is set to one more than the received sequence number, and the sequence number that the server chooses for the packet is another random number.

3. **ACK**: Finally, the client sends an ACK back to the server. The sequence number is set to the received acknowledgement value, and the acknowledgement number is set to one more than the received sequence number.

At this point, both the client and server have received an acknowledgment of the connection. The steps 1, 2 establish the connection parameter (sequence number) for one direction and it is acknowledged. The steps 2, 3 establish the connection parameter (sequence number) for the other direction and it is acknowledged. With these, a full-duplex communication is established.

5.2 Connection Termination:

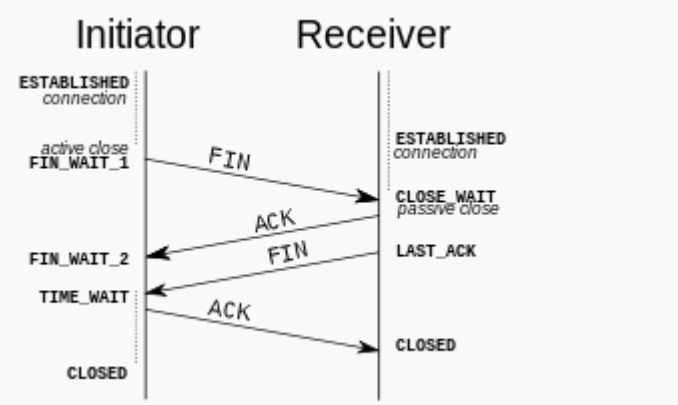

Fig.3: Connection termination

The connection termination phase uses a four-way handshake, with each side of the connection terminating independently. When an endpoint wishes to stop its half of the connection, it transmits a FIN packet, which the other end acknowledges with an ACK. Therefore, a typical tear-down requires a pair of FIN and ACK segments from each TCP endpoint. After both FIN/ACK exchanges are concluded, the side that sent the first FIN before receiving one waits for a timeout before finally closing the connection, during which time the local port is unavailable for new connections; this prevents confusion due to delayed packets being delivered during subsequent connections.

It is possible for both hosts to send FINs simultaneously then both just have to ACK. This could possibly be considered a 2-way handshake since the FIN/ACK sequence is done in parallel for both directions.

5.3 Usecase Diagram :

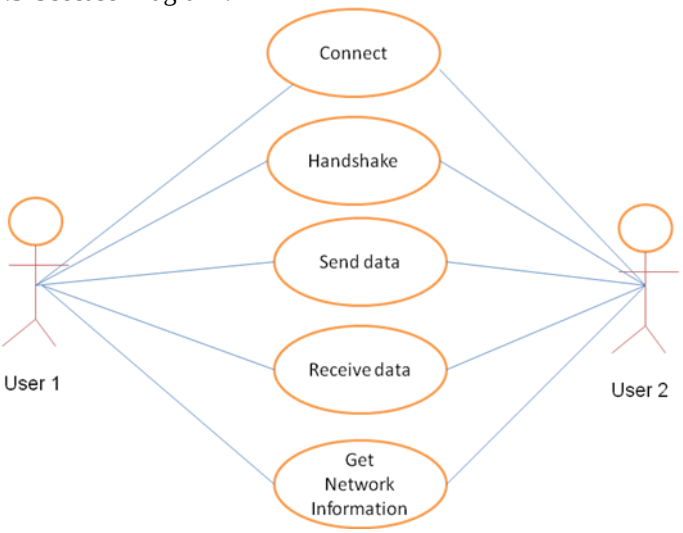

Fig.4: Usecase Diagram of AccessMe

As shown in the above diagram, we are going to require two users i.e. Sender and Receiver. The main use cases are:

Connect: To connect to the network.

Handshake: To perform the three-way handshake, in order to ensure a reliable connection

Send data: To allow the user to send the data.

Receive data: To allow the user to accept the data that have been send to him.

Get Network Information: To display network related details to the user.

5.4 State Diagram:

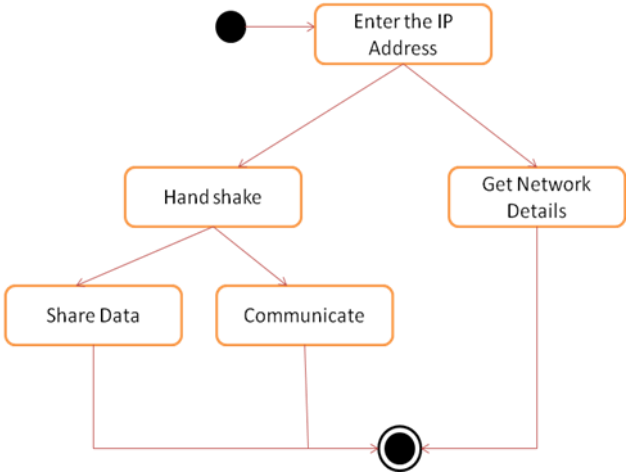

Fig.5: State Diagram of AccessMe

As show in the State Diagram, the basic input required by the application is the IP address.

- If the user entered his own IP address, he will be able to get details of his own network such as latency, number of ports scanned and so on.
- If the user enters the IP address of some another user, he will be required to perform three-way Handshake operation for establishing reliable connection.

- Once the connection has been established, the users can either communicate or share date and resources.
- After performing the required operation, connection can be terminated.

V. CHALLENGES AND LIMITATIONS

Due to the increasing realization of the P2P concepts in the ensemble mobile environment, there are still a wide number of different challenges posed in this particular domain. To sum up, more such challenges are constituted by the mobile environment heterogeneity of (users, devices, networks).Therefore, different requirements should be taken into considerable account for P2P in ensemble mobile environments, and they are highlighted as follows:

• The traffic overhead of P2P lookup needs to be reduced as much as possible, so that mobile devices transmission data rates of can be increased.

• The high churn rates should be addressed because of the open and non-deterministic nature of the nodes which joins and leaves frequently.

• The limitation of mobile device resources and that of nodes heterogeneity and their diverse capabilities of device need to be considered.

• The probability of failure should be well-managed and mobile devices need to be increased because of the breaks of link, discharged energy of batteries.

• The generated traffic in the physical layer by taking into consideration the physical network characteristics.

• Reliable and Trust models should be provided for supporting the users' desires to comply with the rules of protocol.

Thus, it is important to take into consideration these requirements in the proposed mobile P2P protocol for the ensemble mobile environment to overcome the challenges especially the mobility and heterogeneity. These limitations represent parameters that may be adjusted by the protocol programmer to optimize it for a particular mobile device.

VI. REQUIREMENTS

- **Hardware Requirements :**
  512 MB RAM
- **Network Requirements :**
  1. Internet connection(10 Mbps and above)
  2. IP Address and port number
- **Software Requirements :**
  1. Android OS version-4.0.1 and above (for Mobiles)
  2. Blue Stack (for PCs)
  3. Tool: Eclipse or Android Studio(for Development)

VII. CONCLUSION

With this, we can conclude that, AccessMe is one of the best remote controlling software , providing ease of access and lots of features. It has a clean user friendly interface and it provides tools not only for remote access but for network access as well. AccessMe will be available on any platform, from Windows to Android and iOS devices, a feature that can also be found in other alternative software but unlike the competition, AccessMe offers these features at no additional costs.

REFERENCES

[1] KANGWOON HONG, "TELNET: THE MOTHER OF ALL(APPLICATION) PROTOCOLS", INTERNET COMPUTING, IEEE (VOLUME:2), MAY/JUNE 1998

[2] FAWZI BEHMAMM, "TELNET MANAGEMENT", IEEE 2010.

[3] MR.ABHISHEK VERMA, MS.SHILPA VERMA, "WEB SERVER FOR ACCESS WITHIN REMOTE PC'S", IJERA,VNCET 2012.