Scientific Journal Impact Factor (SJIF) : 1.711
International Journal of Modern Trends in Engineering and Research
ISSN (ONLINE) : 2349 - 9745 ISSN (PRINT) : 2393 - 8161

LanLink
Text and Voice communication over wired LAN

Mrs.V.K.Annapurna[1]-Associate Professor, Sandeepa.H.Bhat[2], Vinay Hosamane.K.N[3], Vinit.N.Shanbhag[4],Vinyas.N.R[5]
Department of Computer Science, The National Institute of Engineering, Mysuru, India-570008

Abstract—This paper is influenced by the experiences and designing process involved in developing software for chat over wired LAN where python language has been used as the programming tool. The application can be used to communicate through text and voice methods. The discussion here mainly involves some of the key factors which took part effectively throughout the process such as ARP over NMAP Protocols, protocols and tools used for text and voice communication, dual socket methodology for two way communication and python programming language. Getting the list of computers stored in local ARP cache reduced the time required by the nmap tool. TCP looks after text as well as voice communication in LanLink. Two different port numbers have been used in receiving and sending data both I case of voice and text. Python version 2.7 is the coding language chosen for this network application.
Keywords—*NMAP, ARP, Sockets, Ports, Python, Pyaudio*

I. INTRODUCTION

We are in the age of exploding networks. Networks are nothing but a set of computers interlinked. Benefits of using networks are file sharing, sharing of internet, resource sharing and security. This paper comes out as a part of development of LanLink software which provides the interface for text and voice communication over wired LAN. LanLink is a chat messenger to be implemented in networks like in an office or a college lab. It provides centralized server access.

LanLink has server centric architecture where a central server is responsible for the whole routing of data connecting nodes in a network. The server can be hosted in any of the computers connected to the network where the LanLink network will be hosted. Each client will be initialized by the user by setting up the server address which makes the server portable within the network. LanLink uses TCP for text and voice communication.

This paper is divided into four sections. First section involves the discussion on choosing ARP over NMAP protocols, Second talks about the protocols and tools that have
been used for text and voice communication. Choice of dual socket methodology for two-way communication is explained
in third section. Fourth section describes about the choice of python as a programming language.

SECTION 1: **Choosing ARP over NMAP Protocols**

ARP is short for Address Resolution Protocol which is a network layer protocol used to convert an IP address to physical address.

NMAP [1] is a sort of a tool used to discover hosts and services on a network. This leads to creating a "map" of the network, thus the name nmap. NMAP achieves this by sending specially crafted

packets to the target host and then analyzing the responses. Some of the features of NMAP include port scanning and OS scanning, etc.

In a network based application requiring fast discovery of hosts, using ARP [2] is a better option than using nmap. NMAP runs a sweep of the network by pinging each of the addresses and scanning all ports in them giving details about the states and their usage. It also provides some hardware details. The drawback of it being that in case of certain firewalls installed in some ports it can lead to not being detected by the NMAP protocol. This can hinder the discoverability of a node for some network based application. NMAP also needs a bit of time for ping acknowledgements and as the number of hosts in a network increases it may take longer to scan them. But ARP being one which uses data present in the table (ARP table) saves time required for pings and hence gives a faster response. This ARP table is regularly updated as and when ARP requests are done. The major advantage of ARP is that it works even in the presence of firewalls in a node as ARP is practically a necessity to participate in network communications. One disadvantage being that it doesn't work across networks and can exist only within a network. But this disadvantage becomes trivial when developing applications to be hosted within a network. Thus ARP is found to be more consistently stable and appropriate for applications to be deployed within a network making it a better choice and alternative to using NMAP protocol to discover hosts running a specific application. Figure 1 shows the working of both protocols on command prompt.

SECTION 2: **Protocols and tools used for text and voice communication**

LanLink has text and voice communication methods. TCP is the main protocol used in these communications. Even if TCP has some delay it is reliable. The most important thing TCP gives congestion control: as the links across "sense" the available bandwidth. Same links with UDP will undergo congestion and there will be loss of packets.

In LanLink there is a server in the working at the center of all clients, which takes care of connectivity of clients and routing the messages between the clients. Two sockets have been used with separate port numbers to send and receive the text message messages both in client and server. In the case of routing voice messages same technique is used. PyAudio [3] is used to record the voice for a given interval of time. A file is created and written in wave format and saved with the receiver name and timestamp of recording audio. This file is then transferred to receiver via server. File is also saved in the server. On receipt, a file is created with the chunks of arrived data in wave format. It is saved with time of receipt and sending client's name. Then it is read using Pyaudio play module. With underlying LAN infrastructure working efficiently with good speed can ensure fast transmission and receiving of data packets.

It is observed that for voice communication UDP would serve better as there will not be any acknowledgement to add for the delay. In LanLink the recorded audio isn't sent directly through an open stream to the receiver. But voice recorded and saved as a ".wav" [4] file before it is sent. To send this file TCP services have been made use of again so that reordering of chunks of file will be taken care by the protocol itself.

SECTION 3: **Dual socket methodology for two way communication**

Port number [5] is a local property and helps identify a socket endpoint for an incoming data destined for that port on the receiver's machine. Each machine has 64K ports for each protocol type (TCP or UDP) and for each family type (IPv4 or IPv6).With UDP, it is possible to send to (and receive from) many clients sitting on different ports. With TCP once the connection is established

between two sockets there is no way either of these can connect with a third one. This cannot be exactly called as drawback, as more number of ports are available for communication.

One port alone can be used for two way communication (i.e. send and receive data) with TCP sockets [5]. Because in TCP, the system holds the address [6] as SRCIP, SRCPORT, DESTIP, DESTPORT, PROTOCOL. TCP maintains information like above for every connection. This is not the case in UDP as it is connectionless. In the LanLink two separate ports have been used to set up the two way communication both in case of voice and text data. This methodology solves the main issue of synchronization of data sending and receiving. With poor synchronization mechanisms using single port would cause the connection timed out error on regular basis.

SECTION 3: **Usage of python as a programming language**

[10]With its features of client-server protocol implementations, C integration, and third-party tools and libraries, Python is ideal for prototyping and deploying network applications. Python's runtime interpretation drastically shortens the turnaround from modification to execution, a significant advantage when most changes are incremental and most problems appear at runtime. Python's clean syntax, dynamic typing, integrated exception handling, and object facilities have made it a widely used interpreted language. With the emergence of large-scale-network tools and applications, such as the Twisted framework [7], BitTorrent [8] and Zope content management system, Python has proven to be a legitimate platform for network-oriented projects in a space that C has traditionally dominated.

[9] **Python compared to other languages:**

- **JAVA:** Python programs are generally expected to run slower than Java programs, but they also take much less time to develop. This difference can be attributed to Python's built-in high-level data types and its dynamic typing. Because of the run-time typing, Python's run time must work harder than Java's.

- **JAVASCRIPT:** Like JavaScript (and unlike Java), Python supports a programming style that uses simple functions and variables without engaging in class definitions. Python supports writing much larger programs and better code reuse through a true object-oriented programming style, where classes and inheritance play an important role.

- **C++:** Python it is often 5-10 times shorter than equivalent C++ code. Python shines as a glue language, used to combine components written in C++.

- **PERL:** Python and Perl come from a similar background (UNIX scripting, which both have long outgrown), and support many similar features, but have a different philosophy. Perl emphasizes support for common application-oriented tasks, e.g. by having built-in regular expressions, file scanning and report generating features. Python emphasizes support for common programming methodologies such as data structure design and object-oriented programming, and encourages programmers to write readable (and thus maintainable) code. Python comes close to Perl but rarely beats it in its original application domain; however Python has applicability well beyond Perl's niche.

II. CONCLUSION

The topics discussed under 4 sections were of the outcome of the design methods used to simplify problems faced in the development of LanLink software. These not only simplify the development process but also provide insights to the people who are new to the network communication software designs.

These techniques will fulfill the aim of communication over LAN. It's not just file sharing now but the users will get to connect with many other clients. This will overcome the need of one client from contacting the second client to share some required file(s) by other means (such as calling him over phone or contacting through social media etc.).

Figures and labels:

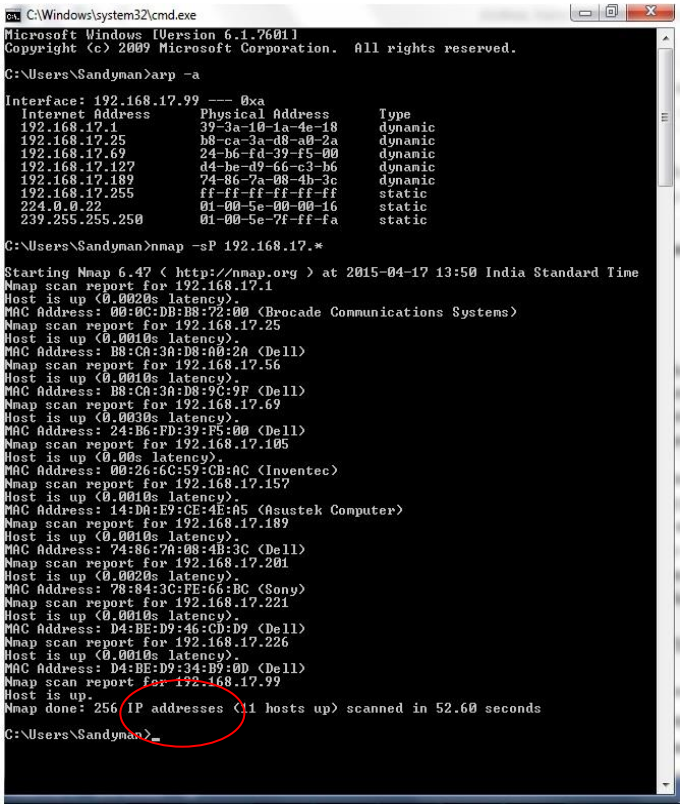

Fig. 1. ARP vs NMAP protocol working

This shows the working of both the protocols on the windows command prompt. With "arp –a" command the list of computers connected can be obtained within no time. But with NMAP tool it requires about 50 seconds. Goes on increasing with number of computers connected over LAN.

REFERENCES

Websites
[1] https://nmap.org/
[2] http://ss64.com/nt/arp.html
[3] https://pypi.python.org/pypi/PyAudio
and https://people.csail.mit.edu/hubert/pyaudio/docs/
[4] http://en.wikipedia.org/wiki/WAV
[5] http://searchnetworking.techtarget.com/definition/port-number and
https://docs.python.org/2/howto/sockets.html
[6]http://stackoverflow.com/questions/11129212/tcp-two-different-sockets-sharing-a-port
[7] http://legacy.python.org/workshops/2002-02/papers/09/index.htm
[8] https://code.google.com/p/python-libtorrent/
[9] https://www.python.org/doc/essays/comparisons/
IEEE Paper
[10] An Introduction to Network Programming: Python Way IEEE
DISTRIBUTED SYSTEMS ONLINE 1541-4922 © 2005 Published by the IEEE