Real Time Network Files Transfer Protocol (RTFTP)

Alaa Noori Mazher
Department of computer and information System University of Technology/Baghdad
Email: laa_19782002 @yahoo.com

Received on: 25/ 4/2012& Accepted on: 4/10/2012

ABSTRACT

In recent days there is a need to keep environment updated. The paper present flexible and simple protocol for updating file system for personal computer through internet. The suggested system depends on the network main component (socket) and uses the User Datagram Protocol (UDP) internet protocol. The system has the ability to gather information through special channel (port) about the used directory. This information includes the files name, files size, file dates, files extension. Through simple protocol this information gives the administrator indicated for the changed file and in what location in order to implement updating the file in the other side. The system serves many users and applications. It provides simple, flexible system that could be installing in any side with few steps.

Keyword: real time, header, file port udp

بروتوكول الزمن الحقيقي لنقل الملفات

الخلاصة

هنالك حاجة ملحة في هذه الايام لابقاء بيئة العمل محدثة. هذا البحث يقدم بروتوكول مرن وبسيط لغرض تحديث نظام الملفات من الحاسبات الشخصية. ان النظام المقترح يعتمد على وسيلة مهمة في الشبكات وهي السوكت وكذلك بروتوكول الانترنت UDP . يمتلك النظام القدرة لجمع المعلومات عن طريق قناة خاصة (منفذ) وهي عن المجلد المستخدم. هذه المعلومات تشمل اسماء الملفات، احجام الملفات، تواريخ الملفات، كذلك اسماء لاحقات الملفات. من خلال بروتوكول بسيط تعطي هذه المعلومات مؤشرات للمدير وذلك عن اي تغير في اي ملف لغرض تحديثة في الطرف الاخر. يخدم النظام العديد من المستخدمين والتطبيقات. حيث انه يوفر نظام بسيط ومرن وسهل التنصيب في اي موقع وذلك بخطوات بسيطة.

INTRODUCTION

The rapid development in communication system and software technologies leads to simple problem. As described (simple problem) but it cause long delay in the work for thousand users. This problem is that there is a great need for simple network handling (for sharing as example), But in the vague and huge amount of network technology we lost the way.

The paper presents simply strategy for real time file sharing across the internet. The paper uses the UDP protocol to handle the transfer. By creating specific folder. The folder is used to contain the shared files, and by using monitoring system to

PDF created with pdfFactory Pro trial version www.pdffactory.com

determine which file had been update. The monitoring system depends on the file properties such as size, modified data and time, and other [1].

RELATED WORKS

"Why File Sharing Networks Are Dangerous", M. Eric Johnson, Dan McGuire, Nicholas D. Willey*, Center for Digital Strategies, Tuck School of Business, Dartmouth College, Hanover NH 03755.

Applications for many users. With millions of users' world-wide sharing music, video, software, and pictures1, file movement on these networks represent a significant percentage of internet traffic. Beyond the much discussed copyright infringement issues, P2P networks threaten both corporate and individual security. Our research shows that confidential and potentially damaging documents have made their way onto these networks and continue to do so. The research also shows that criminals trawl P2P networks and opportunistically exploit information that they find.

"File-Sharing and Copyright", Felix Oberholzer-Gee, Koleman Strumpf, The advent of file-sharing technology has allowed consumers to copy music, books, video games and other protected works on an unprecedented scale at minimal cost. In this essay, we ask whether the new technology has undermined the incentives of authors and entertainment companies to create market and distribute new works. While the empirical evidence of the effect of file sharing on sales is mixed, many studies conclude that music piracy can perhaps explain as much as one fifth of the recent decline in industry sales. A displacement of sales alone, however, is not sufficient to conclude that authors have weaker incentives to create new works. File sharing also influences the markets for concerts, electronics and communications infrastructure. For example, the technology increased concert prices, enticing artists to tour more often and, ultimately, raising their overall income.

"The Effect of File Sharing on Record Sales", An Empirical Analysis*, Felix Oberholzer-Gee Koleman Strumpf, Harvard Business School UNC Chapel Hill, foberholzer@hbs.edu cigar@unc.edu, June 2005.

For industries ranging from software to pharmaceuticals and entertainment, there is an intense debate about the level of protection for intellectual property that is necessary to ensure innovation. In the case of digital information goods, web-based technologies provide a natural crucible to assess the implications of reduced protection because these technologies have drastically lowered the cost of copying information. In this paper, we study the impact of file sharing technologies on the music industry. In particular, we analyze if file sharing has reduced the legal sales of music.

While this question is receiving considerable attention in academia, industry and in Congress, we are the first to study the phenomenon employing data on actual downloads of music files. We match 0.01% of the worlds downloads to U.S. sales data for a large number of albums. To establish causality, we instrument for downloads using data on international school holidays and technical features related to file sharing. Downloads have an effect on sales which is statistically indistinguishable from zero. Moreover, our estimates are of moderate economic significance and are inconsistent with claims that file sharing can explain the decline in music sales during our study period.

PDF created with pdfFactory Pro trial version www.pdffactory.com

" How Reliable is the Oberholzer-Gee and Strumpf Paper on File-Sharing?", Stan J. Liebowitz, University of Texas at Dallas - School of Management - Department of Finance & Managerial Economics, September 1, 2007
This is a lengthy critique of the empirical findings, factual claims, and logic of the empirical examination of file-sharing by Felix Oberholzer-Gee and Koleman Strumpf. It is written for a general audience and provides details of calculations, data, and industry measurements that allow replication by the reader whenever possible. It provides much of the detail for the shorter, more academic version that is also posted**.**

TRANSFER CONTROL PROTOCOL-
INTERNET PROTOCOL (TCP-IP)

The name TCP/IP is derived from the most widely used protocols in the suite: TCP is a byte stream protocol that provides reliable end-to-end communication between two processes running on the same or different host systems; IP is a simple best-effort packet switching protocol that allows many different interconnected networks to share the same virtual address space and form a single internetwork. From the host's point of view, IP makes the underlying internet appear as a single virtual network.

The world's largest computer network, the Internet, uses TCP/IP as its dominant protocol suite. The Internet today is a worldwide internet. It connects over 1.7 million hosts and has a user community estimated at approximately 30 million people. TCP/IP is also used on many corporate networks and is particularly popular with UNIX workstations. However, TCP/IP implementations can be found for almost any class of machine from PCs to supercomputers [2].

USER DATAGRAM PROTOCOL (UDP)

The User Datagram Protocol (UDP) is one of the core members of the Internet Protocol Suite, the set of network protocols used for the Internet. With UDP, computer applications can send messages, in this case referred to as datagrams, to other hosts on an Internet Protocol (IP) network without requiring prior communications to set up special transmission channels or data paths. UDP is sometimes called the Universal Datagram Protocol. The protocol was designed by David P. Reed in 1980 and formally defined in RFC 768.

UDP uses a simple transmission model without implicit hand-shaking dialogues for guaranteeing reliability, ordering, or data integrity. Thus, UDP provides an unreliable service and datagrams may arrive out of order, appear duplicated, or go missing without notice. UDP assumes that error checking and correction is either not necessary or performed in the application, avoiding the overhead of such processing at the network interface level. Time-sensitive applications often use UDP because dropping packets is preferable to waiting for delayed packets, which may not be an option in a real-time system. If error correction facilities are needed at the network interface level, an application may use the Transmission Control Protocol (TCP) or Stream Control Transmission Protocol (SCTP) which are designed for this purpose.

UDP's stateless nature is also useful for servers that answer small queries from huge numbers of clients. Unlike TCP, UDP is compatible with packet broadcast (sending to all on local network) and multicasting (send to all subscribers) [3][4].

PDF created with pdfFactory Pro trial version www.pdffactory.com

Common network applications that use UDP include: the Domain Name System (DNS), streaming media applications such as IPTV, Voice over IP (VoIP), Trivial File Transfer Protocol (TFTP) and many online games.

UDP applications use datagram sockets to establish host-to-host communications. Sockets bind the application to service ports that function as the endpoints of data transmission. A port is a software structure that is identified by the port number, a 16 bit integer value, allowing for port numbers between 0 and 65,535. Port 0 is reserved, but is a permissible source port value if the sending process does not expect messages in response.

PACKET STRUCTURE

UDP is a minimal message-oriented Transport Layer protocol that is documented in IETF RFC 768.

UDP provides no guarantees to the upper layer protocol for message delivery and the UDP protocol layer retains no state of UDP messages once sent. For this reason, UDP is sometimes referred to as Unreliable Datagram Protocol [4].

UDP provides application multiplexing (via port numbers) and integrity verification (via checksum) of the header and payload. If transmission reliability is desired, it must be implemented in the user's application.

Table (1) the UDP headers.

Bits	0 - 15	16 – 31
0	Source Port	Destination Port
32	Length	Checksum
64	Data	

The UDP header consists of 4 fields. The use of two of those is optional in IPv4 (pink background in table). In IPv6 only the source port is optional [5].

Source port

This field identifies the sending port when meaningful and should be assumed to be the port to reply to if needed. If not used, then it should be zero.

Destination port

This field identifies the destination port and is required**.**

Length

A 16-bit field that specifies the length in bytes of the entire datagram: header and data. The minimum length is 8 bytes since that's the length of the header. The field size sets a theoretical limit of 65,535 bytes (8 byte header + 65527 bytes of data) for a UDP datagram. The practical limit for the data length which is imposed by the underlying IPv4 protocol is 65,507 bytes.

Checksum

The 16-bit checksum field is used for error-checking of the header and data. The algorithm for computing the checksum is different for transport over IPv4 and IPv6

PDF created with pdfFactory Pro trial version www.pdffactory.com

[citation needed]. If the checksum is omitted in IPv4, the field uses the value all-zeros. This field is not optional for IPv6.

CHECKSUM COMPUTATION

The method used to compute the checksum is defined in RFC 768:
Checksum is the 16-bit one's complement of the one's complement sum of a pseudo header of information from the IP header, the UDP header, and the data, padded with zero octets at the end (if necessary) to make a multiple of two octets.
In other words, all 16-bit words are summed using one's complement arithmetic. The sum is then one's complemented to yield the value of the UDP checksum field.

If the checksum calculation results in the value zero (all 16 bits 0) it should be sent as the one's complement (all 1's).

The difference between IPv4 and IPv6 is in the data used to compute the checksum.

IPv4 PSEUDO-HEADER

When UDP runs over IPv4, the checksum is computed using a PSEUDO-HEADER that contains some of the same information from the real IPv4 header. The PSEUDO-HEADER is not the real IPv4 header used to send an IP packet. The following table defines the PSEUDO-HEADER used only for the checksum calculation [6].

Table (2) the IP headers.

Bits	0 - 7	8 - 15	16 - 23	24 – 31
0	Source address			
32	Destination address			
64	Zeros	Protocol	UDP length	
96	Source Port		Destination Port	
128	Length		Checksum	
160	Data			

The source and destination addresses are those in the IPv4 header. The protocol is that for UDP (see List of IP protocol numbers): 17. The UDP length field is the length of the UDP header and data.

UDP checksum computation is optional for IPv4. If a checksum is not used it should be set to the value zero [7] [8].

THE PROPOSED PROTOCOL

The System could be summarizing by the following steps:

1.The User 1 need to work on excel office document
2.The User1 want to be connected with the User2
3.The first User creates the directory USRSHR on its personal computer and put the file on it. As shown in Figure (1).

PDF created with pdfFactory Pro trial version www.pdffactory.com

4.The second User2 create the directory USRSHR on its personal computer. As shown in Figure (2).
5.Here the system will check the two directories and try to update them as shown in Figure (3) and Figure (4).

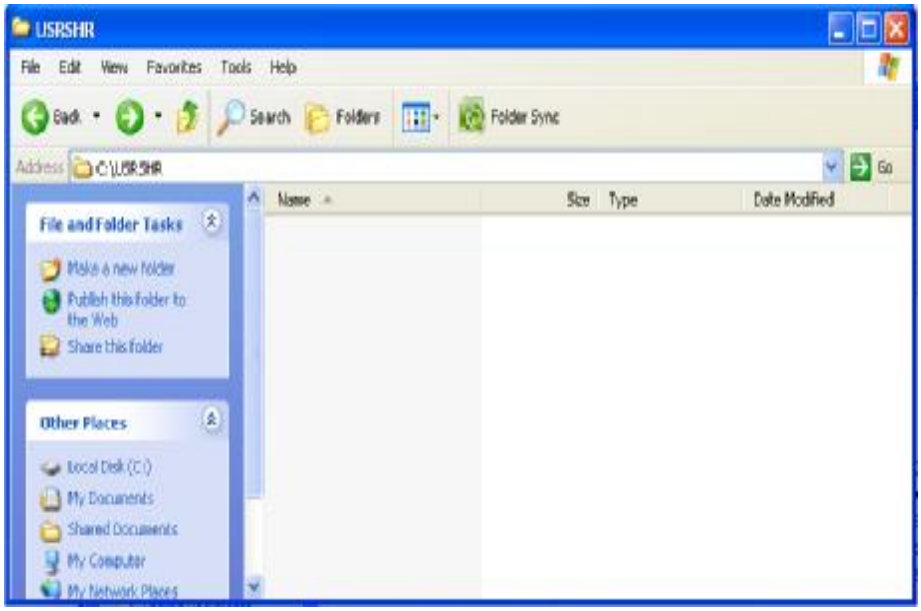

Figure (1): The User1 Shared Directory.

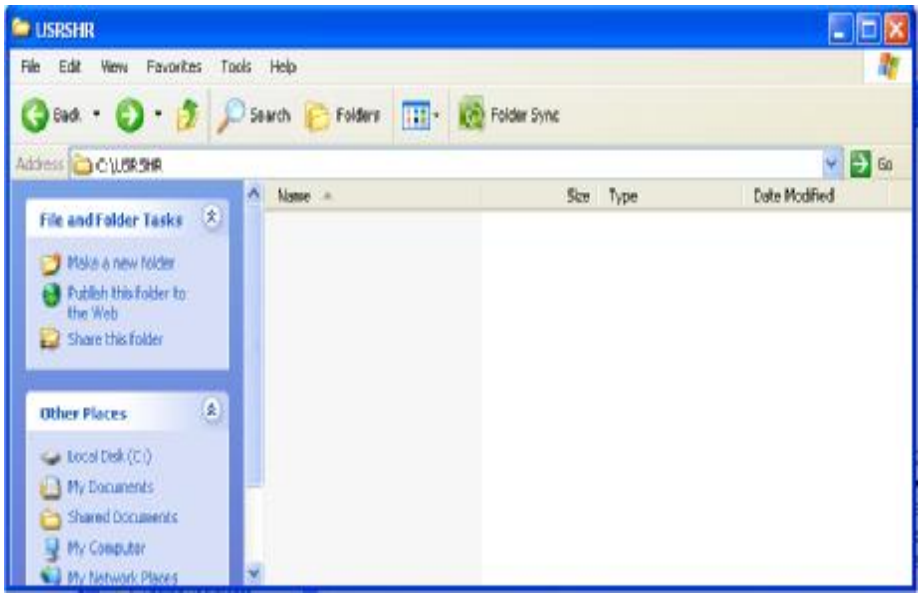

Figure (2): The User2 Shared Directory.

PDF created with pdfFactory Pro trial version www.pdffactory.com

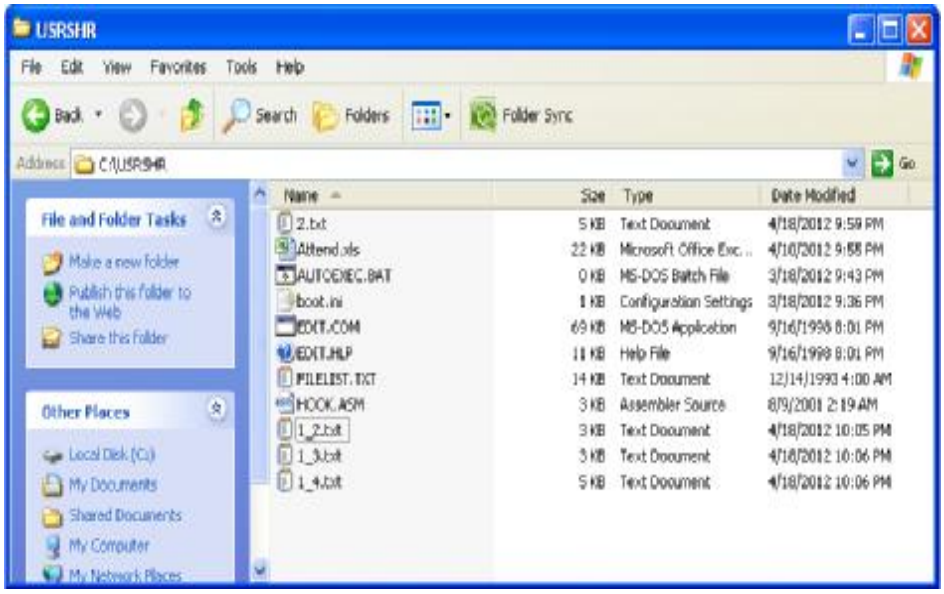

Figure (3): The User2 Shared Directory Filled with Files.

Figure (4): The User1 Shared Directory (after update file 1_4.txt).

From the two figures for the two user directories, the file 1_4.txt has been changed and it is the final updated file depending on the date and time so the system will copy this file from User1 directory to User2 directory.

THE SYSTEM MAIN WORK

a.Copy the missed files on User1 directory from User2 directory.

b.Copy the missed files on User2 directory from User1 directory.Also the system will monitor the files in each directory and recognize if any file change, in order to copy the updated one to other directory for the second user.

PDF created with pdfFactory Pro trial version www.pdffactory.com

The Block diagram in the following figure demonstrates the whole protocol is shown in Figure (5).

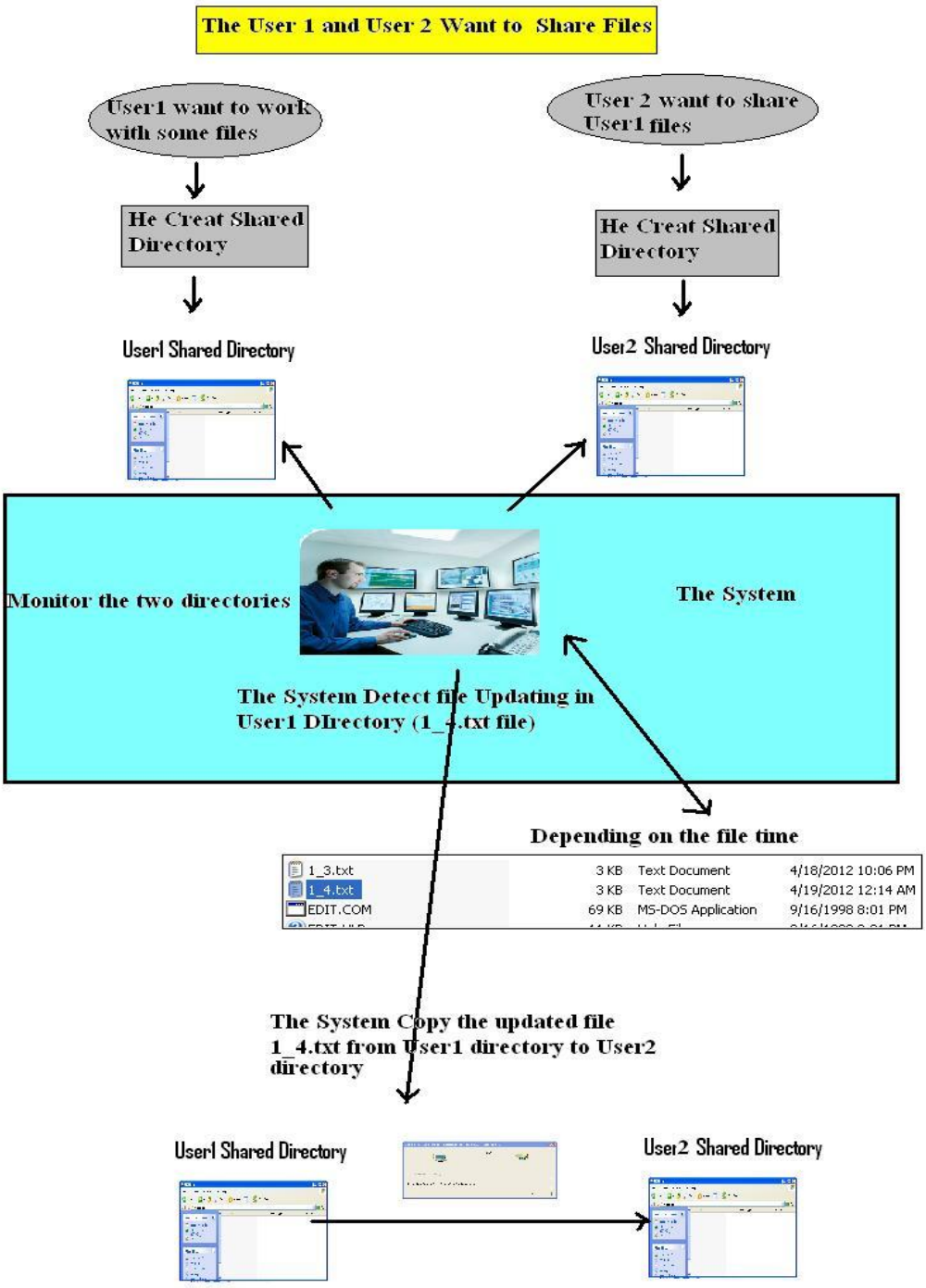

Figure (5): The Adopted System Block Diagram.

PDF created with pdfFactory Pro trial version www.pdffactory.com

RESULT AND IMPLEMENTATION

The system was program using VB and for two sides. The Figure (6) shows the first User (User1) side interface. While Figure (7) show the second user (User2 system interface).

From the figures there are:

1. The File and Dir object while show the shared directory.
2. The Connect to Internet Button use for connecting the system to the internet.
3. The Send Test Button is used for send testes string to the other side.
4. The Reset UDP used to reset the UDP WinScck object.
5. The Update is used to start update process.
6. The Quit Button is used to end the system.

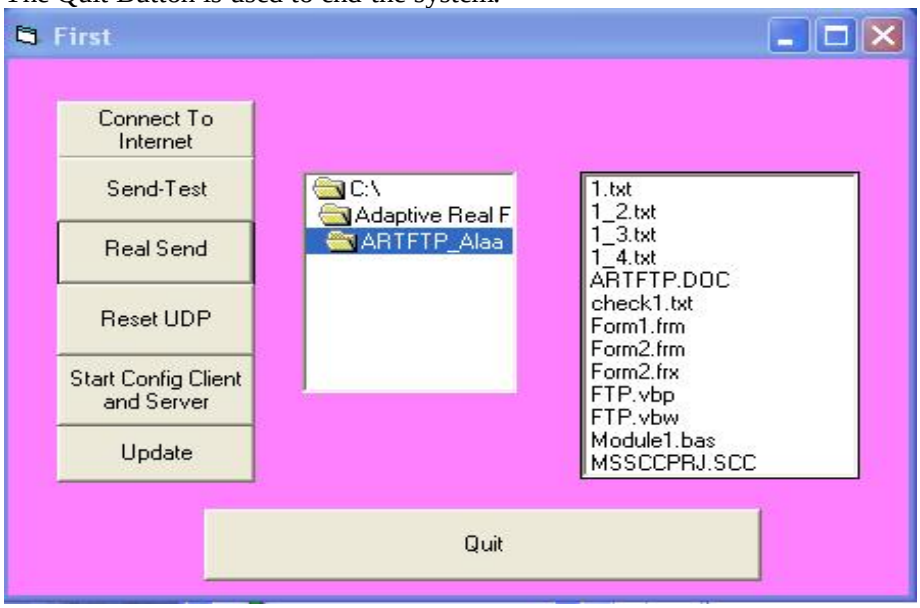

Figure (6): The User1 Interface System.

Figure (7): The User2 Interface System.

PDF created with pdfFactory Pro trial version www.pdffactory.com

DISCUSSIONS AND CONCLUSION

Our paper presents golden solution the simple and very important problem that are arise commonly for many computer users. The Suggested protocol including analyzing the status two location (folder in different location) in order to implement the updating process for the late changing file. It could be applied for varied file type audio, video, text, computer language, office documents, and other. The update process could be implementing periodically as need. Also could be developed in order to apply to the sub folders. Finally the new protocol could be implemented in any Personal Computer.

REFERENCES

[1].Fred Halsall, "Computer Networking and the Internet", 5th edition, Person Education Limited,2005

[2].Behrouz A. Forouzan, "Data Communications and Networking", 4th edition, McGraw Hill International Edition, 2007

[3].Tanenbaum, A. S.; "Computer Network"; 3rd edit; Prentice Hall International Inc; 1996.

[4].Buchanan, W. J.; "Mastering Network"; 1st edit; MacMillan Press LTD; 1999.

[5].Hauden, M.; "Tech Yourself Networking in 24 Hour"; 1st edit; Sams Pub; 1998.

[6].Siebold, D.; "Visual Basic Developer's Guide to SQL Server"; SYBEX; 2000.

[7].Doglas E.Corrclo, "Internetworking with TCP-IP Vol I", Prentice Hall of India, 2000.

[8]. Brien, O'J. A.; "Introduction to Information Systems"; 8th edit; McGraw-Hill; 1997.

PDF created with pdfFactory Pro trial version www.pdffactory.com