VIVA-TECH IJRI
ANNUAL RESEARCH JOURNAL
VIVA-TECH INTERNATIONAL JOURNAL FOR RESEARCH AND INNOVATION
ANNUAL RESEARCH JOURNAL
ISSN(ONLINE): 2581-7280

Peer-To-Peer File Sharing

Pravin Choudhary[1], Mokshada Kesarkar[2], Tejashree Jadhav[3], Archana Ingle[4]
(EXTC, VIVA Institute of Technology, India)

Abstract: *File sharing is a peer-to-peer (P2P) file sharing system that consists of a network of computing devices that work together to form a single system. The implementation of Web RTC for file sharing proved to be effective. The system shares files completely peer-to-peer. It just needs a third peer to act as a signaling server to help these peers to know how to connect to each other. The current stage of the project is capable of sharing file, video/audio streaming between two devices. Multiple devices in a network need more complex management. The system must have no single point of failure and there must not be a need for users to trust each other. The using Web RTC helps to stream real time video and audio buy converting it into a continuous array buffer. Every connection attempt generates a new unique id. The data transferred needs to be converted into an array buffer before sending it and at the receiver peer the array buffered data is converted into blobs. STUN servers are used to get the actual users IP address. Whenever the connection between the two peers is not possible then the TURN (Traversal Using Relays around NAT) server is used for establishing the connection.*

Keywords – *WEB RTC, PEER, SIGNAL CHANNEL, INTERNET CONNECTION ESTABLISHMENT, SESSION DESCRIPTION PROTOCOL*

I. INTRODUCTION

Today's file storage providers are completely centralized and have access to all user's data. Any loss from the company side can lead to user data loss. As there is a tremendous growth seen every year in storage needs, it may happen that we will end up with scarcity of storage devices. One way to solve that issue is to use public devices such as cell phones, pc, etc. that are not going to use their full storage capacity. It can be achieved by a peer-to-peer network of devices to store and share files over the network. There are many peer-to-peer file sharing systems such as Napster, Bittorrent, IPFS and these applications have millions of user file distribution capacities. The upcoming era is becoming much more challenging because of the need for hosting petabytes of data, high quality and large size real-time video/audio streaming, managing versioning of data, and preventing accidental loss of important data. Main motive of this project is to build a distributed file system for sharing and storing data on a network of devices, the system must be fault tolerant and there should not be any single point of failure, the system must be available even if there is any failure and to prevent users data loss by replicating the data.

II. LITERATURE SURVEY

This paper titled “A Blockchain-based Decentralized Data Storage and Access Framework for PingER” published by IEEE Publication in 2018 and the method they have used is metadata are stored on the blockchain whereas the actual files are stored at multiple locations using a PingER Monitoring Agents. There is no single point failure problem since no single machine holds all the data record in the network. The most updated date goes back to 24 hours of time which make it impossible for PingER to capture events in real time [1]. The IEEE-published paper entitled CrowdB C: a blockchain-based decentralized framework for crowdsourcing and its functionality is a system, and the requester's task relies on a third trusted institution. It can be solved by a group of employees without having to do it, the privacy of the user is guaranteed, and the transaction fee is small [2]. In this paper “Cassandra" which is published by Facebook on their own storage systems and how to design them. In this system the distributed storage system for very large amounts of data spread out across many commodity servers. The advantages are Cassandra can support a very high update throughput while delivering low latency and

disadvantages are ability to support atomicity across keys and secondary index support. [3] Paper named "Google File system ". This paper is published by Google on their own storage systems and how to design them. It uses a proprietary distributed file system developed by Google to provide efficient, reliable access to data using large clusters of commodity hardware. The advantages are very high availability and fault tolerance through replication and disadvantages are ability to support atomicity across keys and secondary index support.[4]. The Inter Planetary File System (IPFS) is a peer-to-peer distributed file system that seeks to connect all computing devices with the same system of files. This file system uses Distributed Hash Tables (DHTs) to maintain coordination and metadata about peers. The exchange of data block is done using BiTorrent's data exchange protocol. The version of files is done using Git a Version Control System. It uses Merkel DAG to capture changes in the file system tree. IPFS does self-certification of the files. [5] Raft is a distributed consensus algorithm for managing a replicated log. It solves the issue of getting multiple servers to agree on a shared state even during any system failure If an existing leader fails, a new leader must be elected. Readers need to synchronize the logs of all servers with their own servers through replication. If one of the servers commits a log entry at a particular index, the other servers will not be able to apply another log entry to that index. [6] BitTorrent is a communication protocol for peer-to-peer report sharing (P2P), which permits customers to distribute facts and digital documents over the Internet in a decentralized manner. The BitTorrent purchaser contacts a "tracker" exact withinside the .torrent report. The tracker is a unique server that continues song of the related computers. The tracker stocks their IP addresses with different BitTorrent customers withinside the swarm, permitting them to hook up with every different. [7]The Filecoin network is a distributed, peer-to-peer network formed by Filecoin peers who participate in different ways. Peers communicate over secure channels that they use to distribute information to the network (gossiping), to transfer data among themselves, and to discover other peers, maintaining a well- connected swarm in which information like blocks and messages flows swiftly even when many thousands of peers participate. [8] The network is a Proof-of-Stake blockchain. The blockchain also introduces a multi-level Byzantine Fault Tolerance (BFT) mechanism which aims to provide the same network security as Bitcoin or Ethereum with much higher transactions per second validated. Its multi-level BFT designs for a small set of nodes called a validator committee to produce a chain of blocks as fast as possible while the thousands of other network participants, called guardians, can check this chain's validity. [9]. The paper titled WebRtc peer to peer learning which was published by International Journal of Engineering Research & Technology designs a peer to peer learning system to make the learning more affordable and cost effective which is possible through plugin-free nature of website. It can also be scaled ti serve multiple users simultaneously [10]. The paper - 'Survey on Real time peer to peer multimedia communication application published by Journal of Emerging Technologies and Innovative Research uses WebRtc to inherent features to make it more flexible, effective real-time communication, audio-video conferencing, file sharing and would provide better security of user data and information.[11]. The paper titled File Sharing Strategy based on WebRTC designs and implements a kind of peer-to-peer file-sharing strategy which slices file to blocks on sender browser, and then sender sends each block to a certain node one by one. It ensures the safety of files by P2P transmission, improving the efficiency of the file transfer by the way file is divided into blocks and forwarded. [12]

III. METHODOLOGY

The technology works entirely on a peer-to-peer basis. It only takes a third peer to operate as a signaling server, allowing these peers to communicate with one another. Images, documents, video, and audio files can all be transferred. The flowchart below illustrates how two peers will work together and how they will share files.

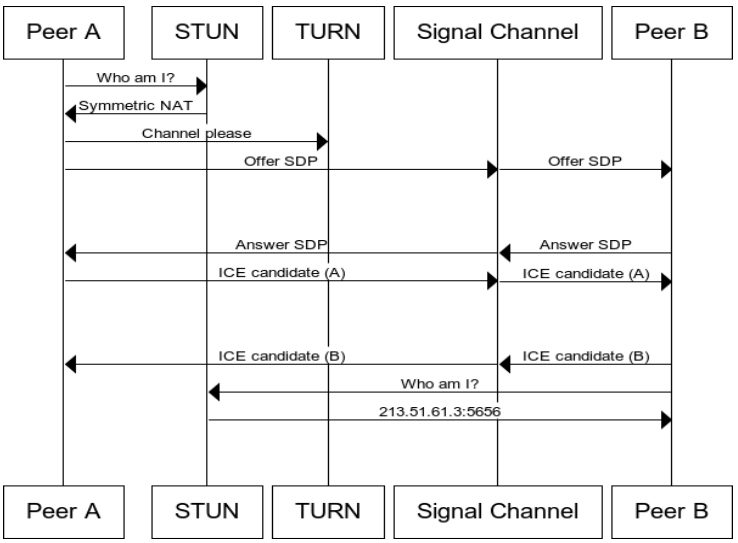

Fig 3.1.Transfer of file between two peers

These are the two users and Peer A and Peer B and the file transfer is done between these two peers. The File transfer can be done such as images, documents, video, audio etc. A STUN server is a server that runs on the public network and replies to incoming requests. The responses it sends out include the public IP address the request was sent to him from. This effectively answers the question "what is my IP address? For most WebRTC applications a server is required for relaying the traffic between peers, since a direct socket is often not possible between the clients (unless they reside on the same local network). The common way to solve this is by using a TURN server. The term stands for Traversal Using Relay NAT, and it is a protocol for relaying network traffic. There are currently several options for TURN servers available online, both as self-hosted applications (like the open-source COTURN project) and as cloud provided services. WebRTC signaling refers to the process of setting up, controlling, and terminating a communication session. When WebRTC applications are said to operate entirely "in-browser," the perspective is taken from the end user's point of view. WebRTC app users require nothing beyond their browsers; but underneath the hood, developers must craft server-side solutions to get peers (i.e. browsers) to communicate with each other. for developers by handling much of the pushing and pulling of data. That relieves app developers of the programming burdens associated with managing versions or locations. They can write the new bits to Firebase and the data will be consistent throughout the system. We used languages such as HTML, JavaScript and CSS for front-end development of the web browser, and JavaScript, Node.js, WebRtc and Firebase for back-end development.

IV. RESULTS

This implementation of peer-to-peer file sharing provides one-to-one file sharing between users while increasing user privacy. Only the sender and recipient peers are involved in the file sharing process, not third party serve Fig 4.1. Connection between two peers
rs. WebRTC works only in browsers, so it has been found to be fast when transferring files peer-to-peer. You can use other libraries like libp2p. The STUN server is used to get the user's actual IP address. If the connection between the two peers is not possible, the TURN server (Traversal with relays around NAT) is used to establish the connection. These results are the following file sharing results between the two peers.

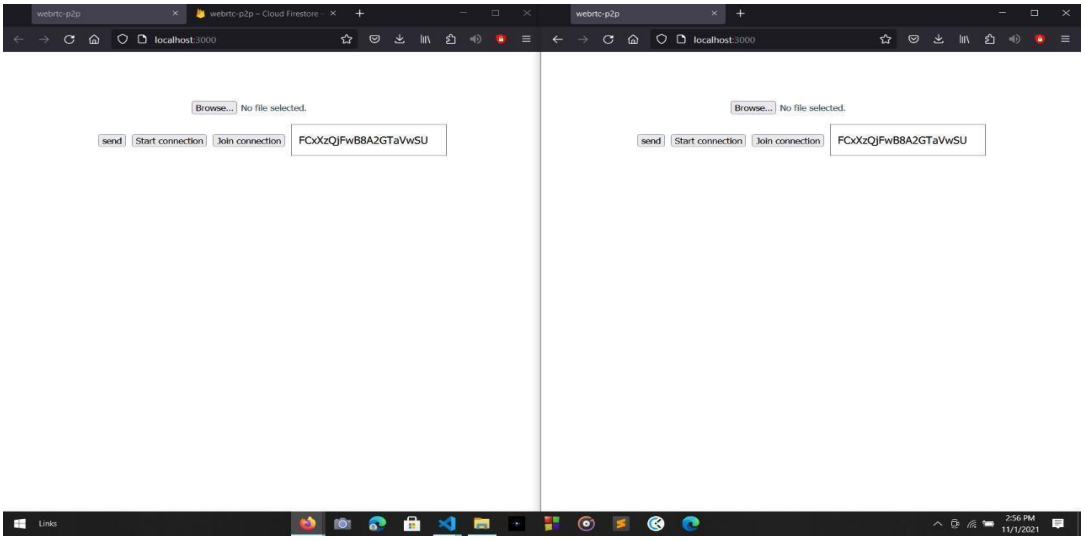

Fig 4.1. Connection between two peers

Peer A has to start a connection by clicking on the start connection button. The browser will communicate with the signaling server and get the generated unique id for the connection. Peer B can join the connection by using the unique id of the connection. Once the offer SDP and ICE are shared the connection between these two peer will be established.

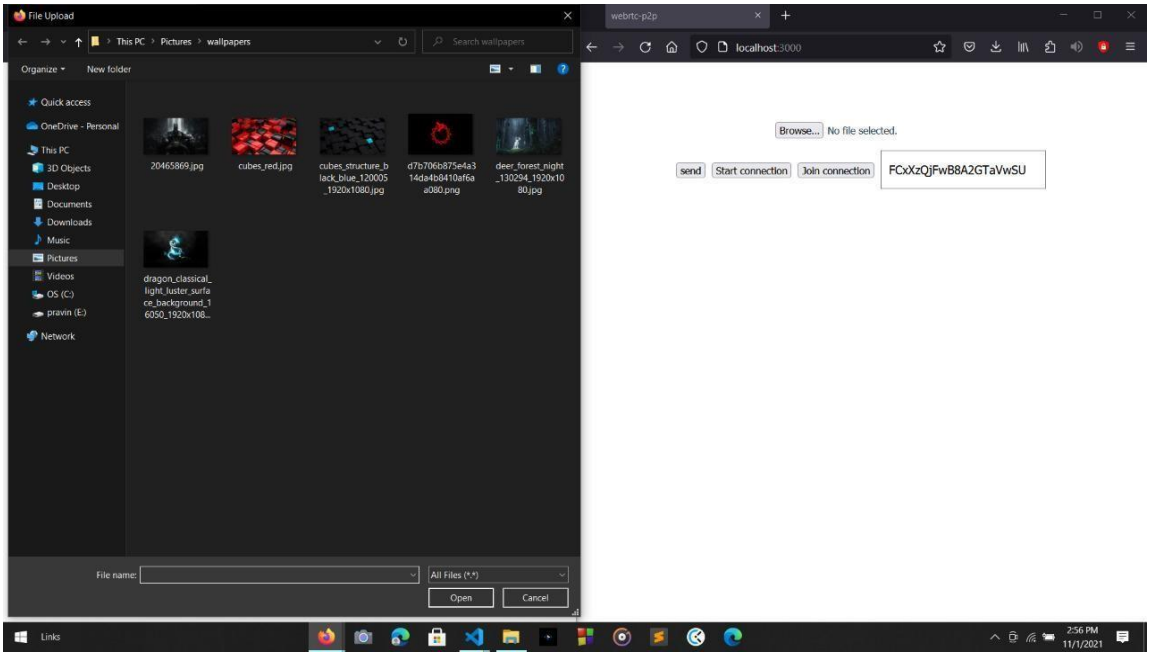

Fig 4.2.Sending File from one user to another

Once the connection is established between users the user can send a file using the input box for selecting the file.

After clicking on the send button the file will be converted in array Buffer and will be transferred from peer to peer using a WebRTC data channel. The data channel is bi-directional. i.e. the peer b can also send files using the same data channel. The received data from the peer A is in the form of an array buffer. The array buffer is converted into a blob. The blob can be downloaded directly from the browser.

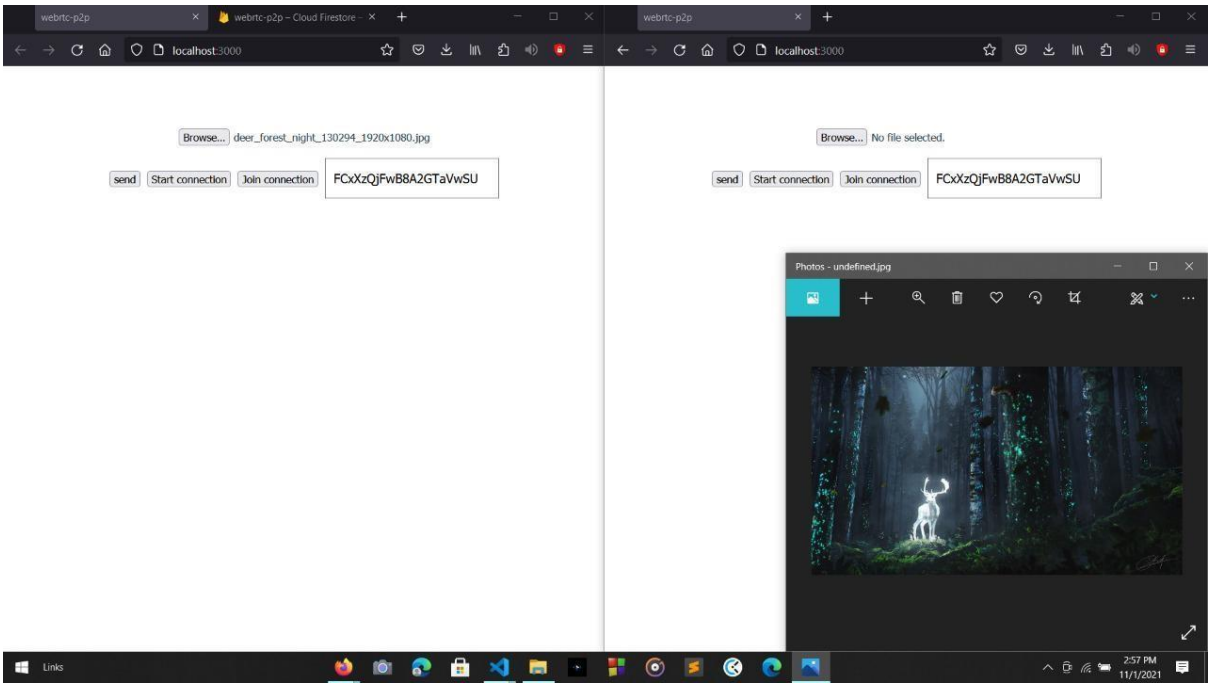

Fig 4.3. Receiving the file and saving the file

The received data from the peer A is in the form of an array buffer. The array buffer is converted into a blob. The blob can be downloaded directly from the browser.

V. CONCLUSION

The file sharing created with WebRTC really works well in transferring any type of file peer-to-peer. It gets difficult to manage multiple peers in a network. This system will always need a third peer to act as a signaling server for connection establishment. For accessing the file stored on another peer the owner peer and the receiver peer need to have the peer Id of both all peers involved in the transaction. During multiple peer networks the discovery of the availability of files located on any peer must be done over the network. In this implementation we used a browser for transferring files but that is not the right way to do it, the system must be accessible from the terminal directly. The storage provider peer must not be able to access or read/write any file of another peer. File data must be kept secured by using a strong encryption algorithm.

REFERENCES

[1] Saqib Ali, Guojun Wang, Bebo White, Roger Leslie Cottrell, "A Blockchain- bases Decentralized Data Storage and Access Framework for PingER". *IEEE Publication,2018*

[2] Ming Li, Jian Weng, Anija Yang, Wei lu, Yue Zhang, Lih hou, "CrowdBC: A Blockchain-Based Decentralized Framework for Crowdsourcing". *IEEE Publication,2019.*

[3]Avinash Lakshman, Prashant Malik, "Cassandra- A Decentralized Structured Storage System". *Facebook,2009.*

[4]Sanjay Ghemavat, Howard Gobioff, Shun-Tak Leaung. "The Google File System" . *Google,2003.*

[5]Juan Benet. "IPFS- Content Addressed,versioned, P2P File." *Facebook,2009*

[6]Diego Ongaro, John Ousterhout. "In Search of an Understandable Consensus Algorithm". *USENIX Annual Technical,2014.*

[7]BitTorrent Publication. "BitTorrent (BTT) WhitePaper." BitTorrent Publication, *February 2019.*

[8]Protocol Labs. "File Coin- A Decentralized Storage Network". *Protocol Labs,2017.*

[9]Theta Labs. "Theta Mainnet 3.0 WhitePaper". *Theta Labs,2020.*

[10] H. Fateh Ali Khan, A. Akash, R. Avinash, C. Lokesh, "WebRTC Peer to Peer Learning". *International Journal of Engineering Research & Technology (IJERT) Vol. 7 Issue 03, March-2018*

[11] Sanket Jadhav, Manoj Berad, Tejas Banaitkar, Swapnil Salunkhe, Swati Khokale, "Survey On Real-Time Peer To Peer Multimedia Communication Application" *JETIR April 2021, Volume 8, Issue 4*

[12] Quanfeng Duan, Zhenghe Liang, "File Sharing Strategy based on WebRTC", *2016 13th Web Information Systems and Applications Conference*