II. Literature Review

To adequately prepare for the coming solutions discussed in this paper it is necessary to go over several primary technologies this paper relies on. These being Blockchains and The Onion Router Project (Tor)

Etherum is a distributed network of computers that manages its resources using cryptography and proof of stake algorithms. Resources are managed via the transactions that are signed off on using cryptographic techniques that verify their authenticity. The method that the Ethereum network uses is known as a Blockchain. Blockchains are ledgers of information that are tied to particular account known as "wallets". Each time this ledger is updated all computers on the network verify the authenticity of the update with proof of stake algorithms. Computers are motivated to update the ledger via payments of a token called "gas". When a transaction is made the user who initiated it will pay a small fee in gas depending on the size of the transaction and the load the network is under. The nature of the block chains cryptography allows for near absolute certainty of the validity of a transaction. Another well known blockchain is Bitcoin, which differs from Ethereum in that is can only store information relating to how much bitcoin a wallet owns. Ethereum allows for the storage of many different types of information such as code, which the network is able to use to run decentralized apps. This ability to store predefined behavior that operate securely on a distributed network is why it is the choice blockchain for this paper. The major downside of Ethereum however is its greatest strength. Due to the open nature of the ledger every computer on the network is able to see all the data in the ledger. Transactions can also be limited based on how much stress the network is currently under as well. To complement Ethereum the second technology that will be discussed below will be implemented.

Tor is a networking protocol that anonymizes internet traffic. It works via encrypting data which is then sent through the tor network. The network is made up of many peer to peer connections that bounce your data around preventing any one sever from knowing where your data is going and where it came from. Tor is an open source project that was originally created by the US navy to hide secret communications. Since then it has seen wide adoption by the greater public for accessing network resources like websites anonymously. The core interest of this project in tor is its ability to host resources anonymously. Using Tor nodes as rendezvous servers (Severs that allow clients behind network address translators to communicate) it is possible to host servers located behind firewalls. This secure true peer to peer behavior would allow each user on a communication network to be their own message server. Combining these two technologies allows for decentralized anonymous network traffic that can be directed by a decentralized authority that can verify users securely and anonymously. Several other projects have attempted to create similarly secure apps which will be discussed below

bChat is a decentralized, immutable, secure chat application. The core structure of many chat apps is completely centralized. Storing and forwarding messages on centralized company owned servers. This is a issues due to the ability for these companies to eavesdrop on conversations and censor content. The centralized nature of this system acts as a single point of failure. If the main servers go down the users will be unable to communicate. bChat attempts to solve this issue by storing all user information on the blockchain. This is near true peer to peer as there is no middle man besides the decentralized Ethereum network. However, due to the nature of the network, this presents several limitations. First, Depending on the current traffic on the network transactions can take some time or be more or less expensive to send messages preventing fast messages. Second due to the nature of the blockchain this would only allow for the sending of text messages, any other media would be unfeasible do to the prohibitive cost and latency. The next project solves the media limitations and latency issues of this implementation while sacrificing some features.

This paper never named their project so for ease of communication we will simply refer to it by the window label present in images from the paper, **CS244B**. The main purpose of this article was to describe how to implement a decentralized messaging application. By leveraging a the matrix protocol, a protocol which describes how servers and clients should talk to each other, CS244B is able to create a decentralized plat from where users can host their own servers. It does not rely on the blockchain and directly routes information to clients through the self hosted server. This solves the media issues as regular networking allows for all manor of media. However it sacrifices several features such as blockchain user verification, immutability, and the ability to work without setting up a home server. This prevents users that are less tech savvy from using this implementation as setting up and managing your own server can be daunting for many inexperienced tech users. This paired with the lack of message verification makes it unideal for mass use. The following project make this distributed architecture but solves the issue of usability.

D.I.M Or Decentralized Instant Messenger is a software that uses the GunJs framework to implement a distributed messaging app. GunJs provides tools to create distributed databases (DDB) across many nodes. This allows for network of self hosted server nodes that store user information. Client nodes on the network are able to read information from the DDB which allows for user verification and logins from any device. The main benefit of this model is that it is easily used by users who just need to install the app or access the web portal. However, This structure lacks several features due to the server node client node model, it is not truly peer to peer even though it is distributed. However this model has no single point of failure communication besides access to the web portal or app and uses encryption methods to maintain and ensure transactions on the DDB are legit. Making it secure and easy to use. The Final project that will be discussed in this literature review has the most similar structure to the proposed solution of this paper.

Muhammad S. A.[] proposes a solution not for a chat app but a way for patients to transfer data from smart sensors to their doctors securely and without a middle man. In [] two technologies are focused on to allow for a secure peer to peer

architecture. [] states that they use the Ethereum block chain for "Medium for negotiation and record keeping" and that tor is use for "Delivering data from patients to doctors". This architecture is favorable for the medical setting due to its ability keep data in the hands solely patient and the doctor. [] uses the blockchain for identity management [] is able to securely transfer the data between doctor and patient using tor rendezvous servers. This is true peer to peer communication and since its using tor it has the added benefit of hiding user metadata such as origin IP.