**Introduction**

Concerns about privacy seem to be getting a lot more attention from the media in recent years. There have been many reported cases of companies spying on users in their networks, so this should come as no surprise. PRISM stands out amongst cases of spying, as here the NSA collaborated with companies like Google and Microsoft to collect data from their unsuspecting users. Other companies, such as Facebook, have been accused of buying and selling user information. The issues presented here could also be important among people who work at corporations as well, considering packet sniffers or other external programs can be used to intercept messages and identify information about them. Our Anonychat will attempt to resolve this need by implementing a system in which packets can neither be read from external programs nor traced with certainty.

Several protocols already exist that provide anonymous chat. For instance, the Invisible Internet Project (I2P) is an ongoing effort to build a free, open source, and anonymous internet. However, I2P is designed at the network layer, and therefore restricts its users by forcing them to use the I2P network. Freenode is another protocol that uses encryption to hide the identity of a user who sends packets over the network. That being said, Freenode uses direct connections that allow its senders to be traced. Competition also extends to the peer-to-peer style communication network Skype, which uses a similar connection system we intend to implement (a central server to start, then peer-to-peer communication afterwards). However, Skype’s main focus is not anonymity, and its peer-to-peer connections use a direct method. Therefore, any anonymity the system does provide could cease to exist later. Other projects are in development squarely to accomplish anonymity; Quassel and Rust are two such examples. However, neither of these projects use the peer-to-peer model we will attempt.

To compete with these various systems, we have implemented a peer-to-peer chat system. Clients ping a central name server to determine and connect to their peers. Once connected, users can communicate with anyone else on the channel. Anonymity is the central focus of our protocol, providing us with an angle different than most of our competition. Second to anonymity is terms of importance is practicality; our protocol must provide a rapid means of communication without saturating the network to be truly successful. We maintain anonymity by broadcasting all sent messages and using two layers of encryption in a way similar to onion routing. When receiving messages, users attempt to decrypt it using public-private keys, and rebroadcast the message regardless of success. We maintain practicality by illustrating that a large network of clients can be sustained without suffering excessive slowdown or consuming too many resources.