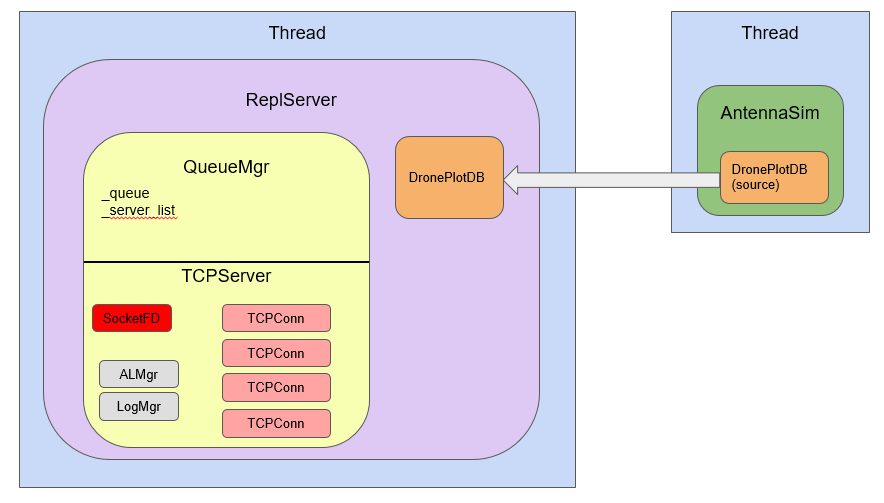
Mark Demore, 2d Lt

CSCE689 – HW4

**CODE:**

<https://github.com/mdemore2/AFIT-CSCE689-HW4-S>

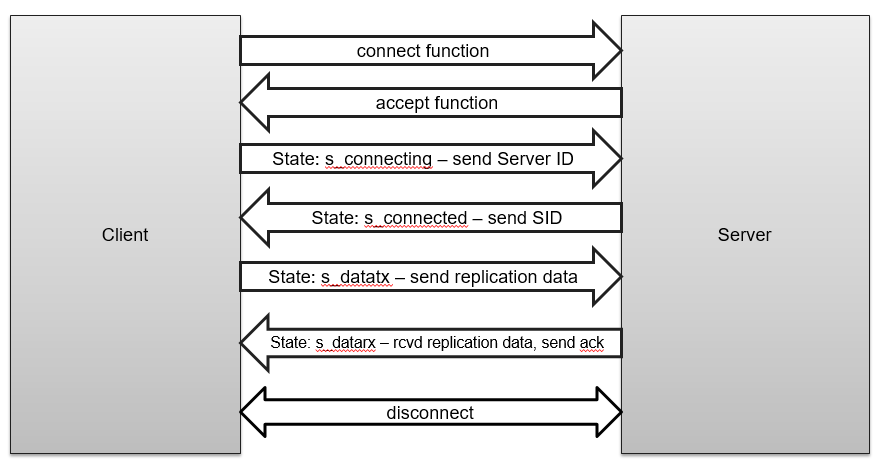
**QUESTIONS:**



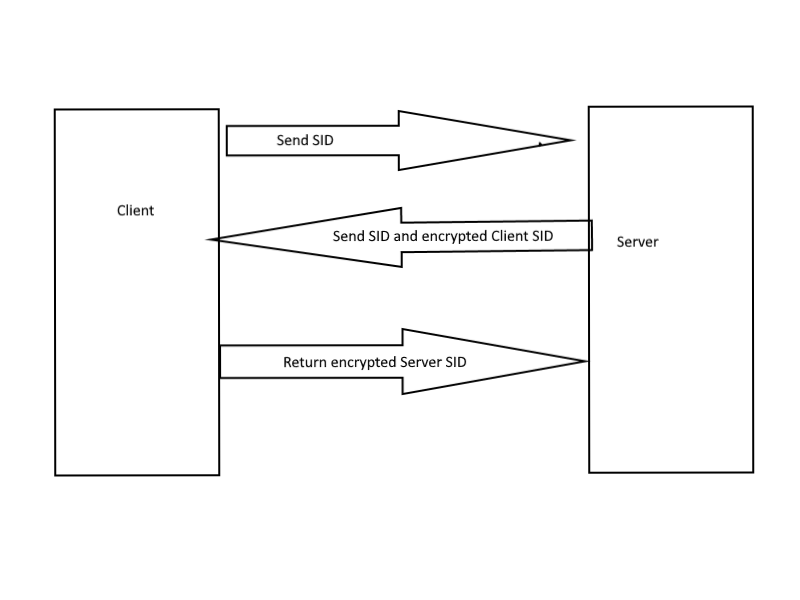
**Figure 1: System Architecture**

The image above shows the basic architecture given for this assignment. The consistency model for this system makes use of the queue manager to pass new plot points between the other servers to be replicated and then deconflicted for time skew and duplicate points. The system uses basic TCP communication protocols handled within the TCPConn class using Socket File Descriptors. Messages are passed using AES Encryption after authentication, shown in the images below. The authentication steps occur in conjunction with the s\_connecting and s\_connected states, making use of the new states s\_authenticate and s\_handshake. Reflection attacks are protected against with a simple check that the client/server is not sent its own SID by the other client/server to encrypt. Data is deconflicted on the replication server with a series of checks. The points are looped through and compared. If the two points have the same location and one of the points came from the elected leader or its time skew has already been calculated, the time skew is calculated for the other node. If the skew could not be calculated yet, it was ignored and revisited again once replicate() was called. Once the skew has been detected for all nodes, the times are adjusted for all points of those nodes in the database. Then, the points are checked for duplicates. If two points have the same timestamp after the skew correction, the point from the node with the lower priority is deleted. The leader is selected simply using a priority queue of the existing servers. This assignment makes use of a client-consistency model, with each client being responsible for maintaining its own data.

All of these design decisions were selected based on the provided architecture and the inherent abstraction of the system in this assignment. This also made the project a bit difficult to understand and interesting in assessing what work needed to be done. It was also difficult to debug and interpret output.



**Figure 2: Original Communications Protocol**

****

**Figure 3: Authentication Steps**