Latimer-Zerachiel System

Ethan Joachim Eldridge

January 2, 2011

0.1   Overview

The *Latimer-Zerachiel* System is an attempt by myself, to create a sort of remote access program for two computers linked over a network. The configuration for the system is as follows:

1. Two Computers are connected via LAN

2. One Computer runs the *Latimer* System, the other the *Zerachiel*.

3. The two computers agree on a specified port and *Zerachiel* connects to *Latimer*

4. The user is then free to send commands via *Zerachiel* and these commands will be executed by *Latimer*

As you can see, it is a fairly simple program when broken into it's higher level model. However, this is a good coding project because of the following difficulties:

1. You must write a program that will link two computers over a network

2. *Latimer* must recieve character data and be able to parse this data to figure out what to do

3. *Latimer* must run the given commands correctly, no matter what OS

4. *Latimer* and *Zerachiel*must handshake and be able to access each other freely

5. You must be able to shut down *Latimer* from *Zerachiel* via a kill code.

The core modular of each of the listed items will be explained further in the following sections.

0.2   The Parser

The parser takes two command line arguments when called, The first, is the name of the parent calling it, and the arguments to be executed. The parent program determines what happens with the arguments. If *Latimer* calls the program, then the aguments are fed through the parser and a string to be passed to the host computer system is created. If *Zerachiel* calls the program, then the argument must be the correct handshake code to begin transmitting data into the Latimer system. If the parser rejects the handshake, then *Zerachiel* will disengage the socket and reject *Latimer*.

Latimer is the main program that will call the parser extensively. This might seem that it would be wise to hardcode the parser itself into *Latimer*, however to keep the program modular, and to test my own skills at batch

programming, the parser is a seperate program. Also, the parser, although this has not been decided, may read commands to be used from a txt file, however, as of this beta, that feature will not be implemented, or at least not until the Entire Prototype is complete.

Once the modular commands are in place, different operating systems, and different configurations for security can be used. And if the user of *Zerachiel* would like to add in functions and commands, then they can use Zerachiel itself to open and edit the commands file hosted on the *Latimer* system.