Backdoor

The purpose of the backdoor assignment was to create a silent, invisible backdoor program that gives an attacker unauthorized and unknown access. An always-on Python listener waits for a TCP connection from a machine running the C# backdoor program. Once connected, the backdoor starts a key logger and presents the listener with a command line interface. Furthermore, the listener has the ability to send and receive files to and from the backdoor machine. If the attacker is finished with the backdoor machine, the attacker can send the exit command to shutdown the backdoor.

The major components of the assignment were the socket programming, the command line interface, and the key logger. By far, the socket programming was the most difficult part. In the first version of the application, the UDP protocol was used and later had to be changed to the TCP protocol. The other major problem with the sockets was the loss of new line characters being sent across the connection. The solution was to split all strings on the new line character and send each string individually.

As for the key logger, an open-source project was found on www.codeproject.com that provided a hookless key logger for C#. The proper attribution is found in the keylogger C# source code file. The keylogger runs in a separate thread and saves all keystrokes to a file every five minutes. The keylogger works flawlessly with the seldom exception of throwing an IO exception stating that the keylog file is open in another process. The exception cause could not be tracked down and is simply thrown out since the logger still works. When the keylogs are sent back to the listener, they are zipped using the Ionic zip library found at https://dotnetzip.codeplex.com - proper attribution is given in the Main C# file.

Finally, the command line was incredibly simple using the Process class in C#. The output can be captured using an event handler and sent back through the socket; the error output is handled in the same manner. When a command is received, a StreamWriter is used to supply a command to the process.