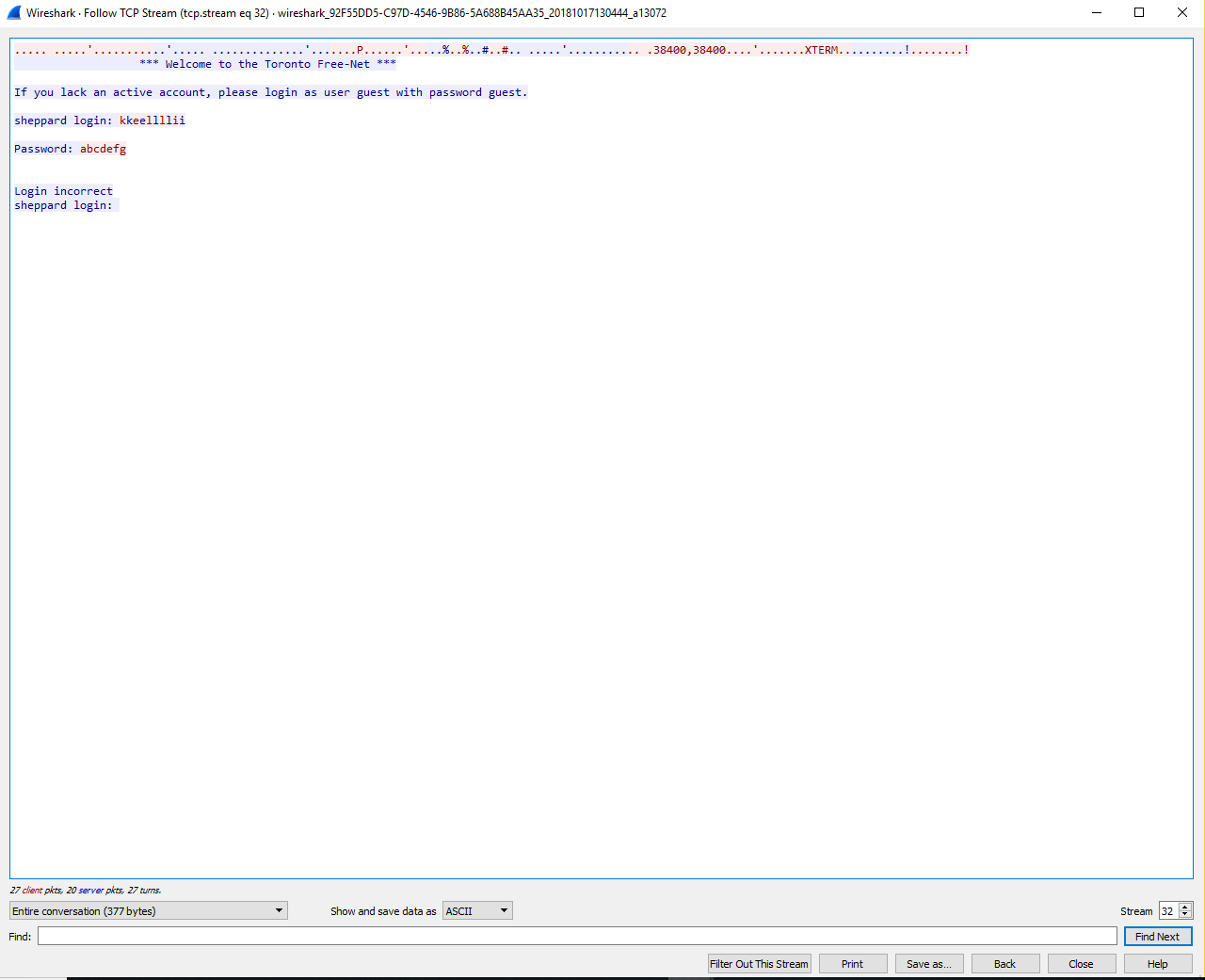
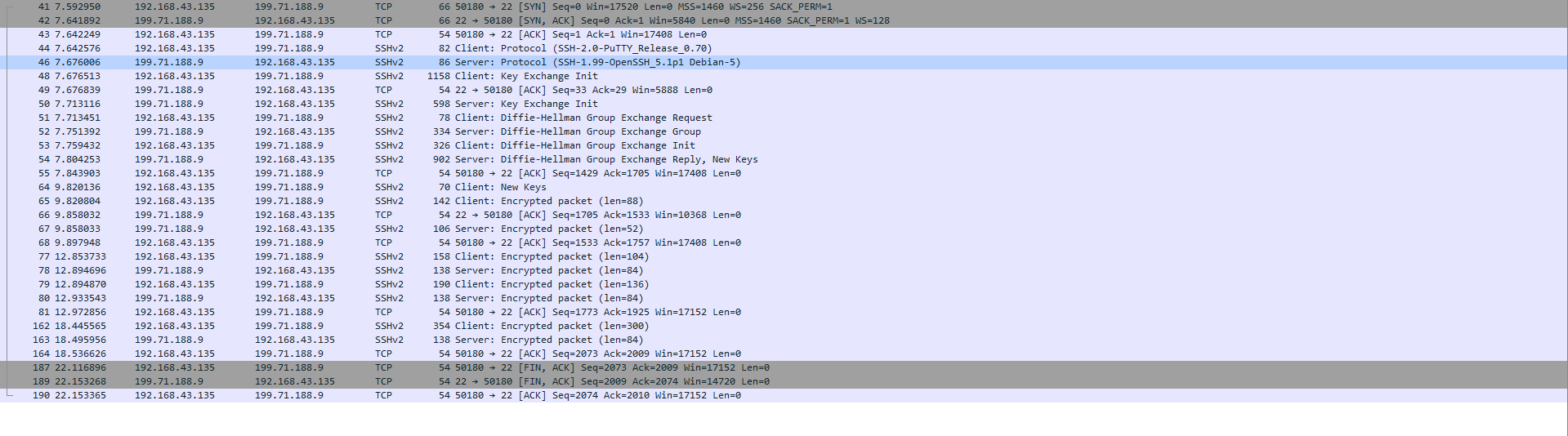
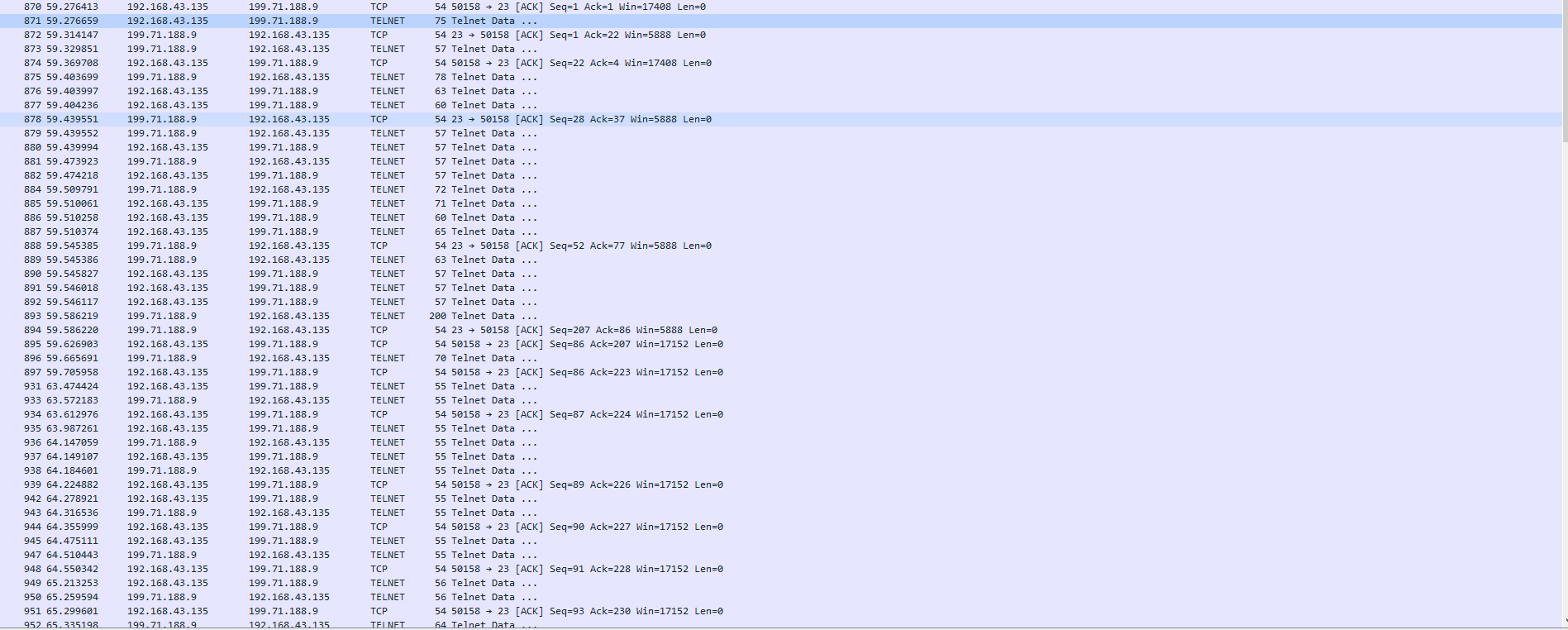
g.) **This screenshot shows what the console looked like during the putty session. The areas highlighted in red represent the input from the user. The “kkeellllii” part is highlight in alternating red and blue reflecting the input from the user being echoed back onto the console. The blue highlighted text is data printed to the console. The biggest thing I notice from this method is that if someone is using telnet to log into a service, and there is an attacker capturing the packets, it appears that they might be able to gain access to the username and password of the person logging in.**

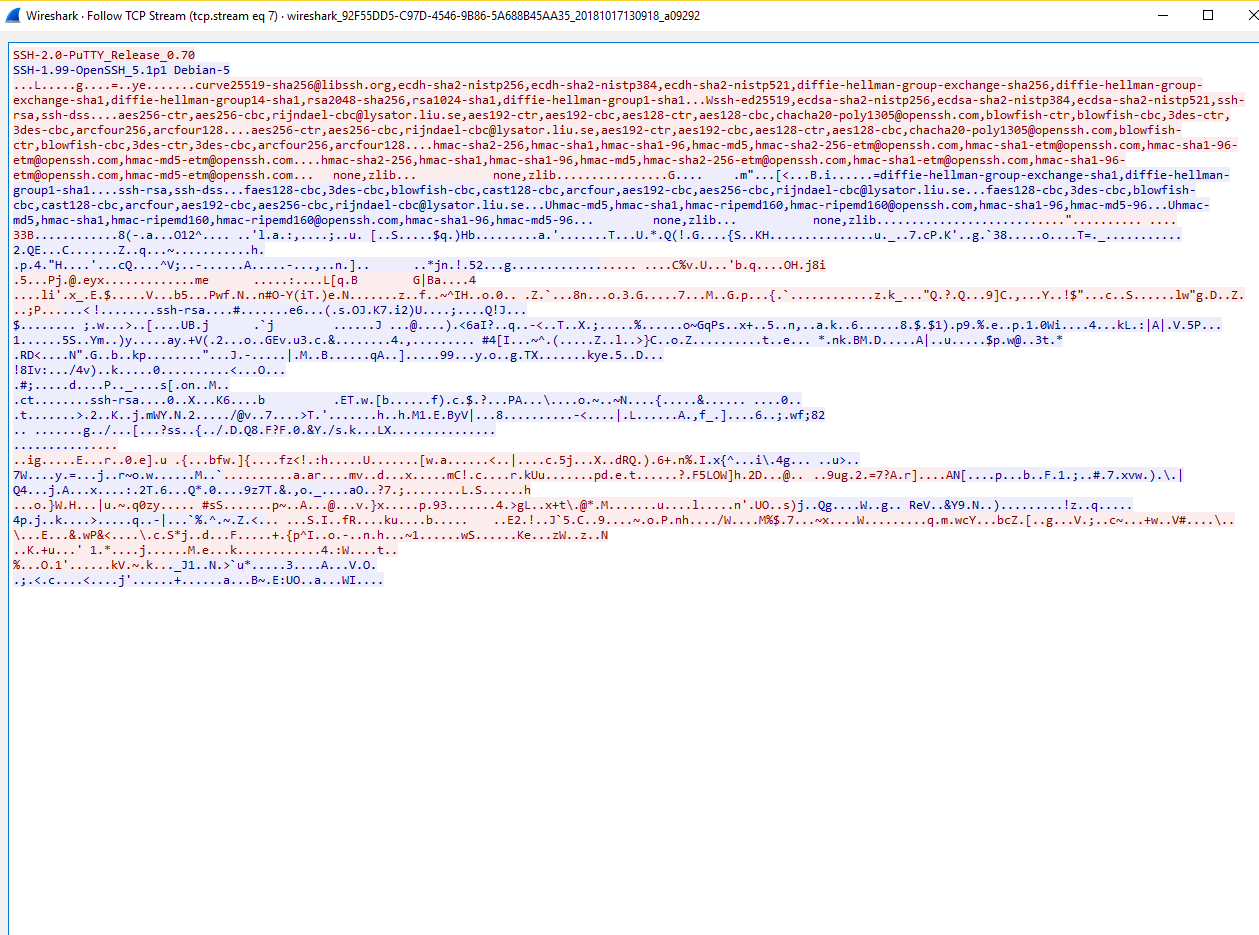


**The screenshots below represent the actual packets captured. You can see what the origin of the telnet data is by the differing IP addresses in the second column. The destination is then the address in the third column.**





**The next step was to attempt the same thing, but this time use SSH in which I received the following packets. You can’t see any accurate representation of what goes on between the console and the server. Notice the password and username aren’t nicely outline in red for anyone to read who can conduct a packet sniffing or a man in the middle attack**.



**The next screenshot shows the actual process that the SSH protocol did when establishing communications with the server. The process starts with the “Server: Protocol (SSH-1.99-OpenSSH\_5.1p1 Debian 5)” and then continues down for the different Key exchanges, including a Diffie-Hellman key exchange that we have discussed earlier in this class. The difference below, is that all information is encrypted and I was not able to view my username and password attempts. When analyzing the server to client and vise versa communications, I see “Encrypted Packet” indicating that I cannot view any of them information being communicated.**

