Northeastern University

Department of Electrical and Computer Engineering

# EECE 4638: **Wireless Design and Simulation**

Lecturer: ***Rahman Doost-Mohammady***

TAs:

**Subhramoy Mohanti**

Homework # 1 : Socket Programming

Group:

*Taylor Skilling (001171439)*

*Chris Kenyon (001112428)*

Semester: Summer II 2015

Date: July 27th, 2015

Lab Session: *Monday-Thursday, 3:20 PM*

##### Lab Location: 429 Dana Research Lab, Northeastern University, Boston, MA 02115

##### Content Page

##### Introduction / Objective 3

##### Design Approach 4

##### Results and Analysis 5

1. **Introduction / Objectives**

In order to better understand socket programming and TCP and UDP communication protocols, a few server/client applications were created and simulated using C++ and NS3.

Unlike TCP, UDP does not implement flow control or error checking. TCP protocol is the most widely protocol used as it offers error correction, preventing the reception of new packets until the previous ones are successfully transferred. UDP is typically used in streaming audio and video as an end-user is more willing to accept frame errors in a video than a slow loading time.

1. **Design Approach**

Using the *sys/socket.h* library and *ServerSocket* and *ClientSocket* clases, direct echo and relayed echo applicatons were created using both aforementioned protocols. The direct echo applications creates communication between a single server and client pair. The delayed echo instead relays a message from a client through a server to a destination, before returning the message back through the server to the original client. Clients and servers in the applications ran on different computers within the Northeastern College of Engineering network in order to take a variety of timing measurements. The difference in speed between UDP and TCP protocols was analyzed and compared to the the software utility *ping.*

1. **Results and Analysis**

The results of the TCP/IP and UDP applications are shown below in Figures 1 and 2, respectively.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **TCP/IP** | | | | | | |
|  | **Relay Echo** | | | | | |
| Client | ohms | | laminar | | laminar | |
| Server | nano | | nano | | nano | |
| Destination | laminar | | ohms | | hertz | |
| Trial Type | User | ping | User | ping | User | ping |
| Trial Number | Time (microseconds) | | Time (microseconds) | | Time (microseconds) | |
| Trial 1 | 854 | 293 | 934 | 353 | 739 | 316 |
| Trial 2 | 712 | 349 | 737 | 325 | 765 | 297 |
| Trial 3 | 781 | 308 | 821 | 355 | 771 | 302 |
| Trial 4 | 833 | 326 | 687 | 344 | 879 | 301 |
| Trial 5 | 973 | 311 | 648 | 310 | 815 | 316 |
| Trial 6 | 805 | 329 | 814 | 344 | 808 | 308 |
| Trial 7 | 946 | 306 | 825 | 304 | 862 | 299 |
| Trial 8 | 770 | 325 | 777 | 319 | 774 | 252 |
| Trial 9 | 794 | 304 | 810 | 350 | 858 | 295 |
| Trial 10 | 693 | 308 | 737 | 337 | 803 | 319 |
| Average: | 816.1 | 315.9 | 779 | 334.1 | 807.4 | 300.5 |
|  | **Direct Echo** | | | | | |
| Server | nano | | nano | | nano | |
| Client | ohms | | laminar | | hertz | |
| Trial Type | User | ping | User | ping | User | ping |
| Trial Number | Time (microseconds) | | Time (microseconds) | | Time (microseconds) | |
| Trial 1 | 250 | 266 | 203 | 273 | 312 | 226 |
| Trial 2 | 244 | 311 | 249 | 305 | 319 | 268 |
| Trial 3 | 288 | 317 | 308 | 261 | 170 | 270 |
| Trial 4 | 200 | 293 | 271 | 272 | 335 | 246 |
| Trial 5 | 291 | 281 | 313 | 302 | 272 | 278 |
| Trial 6 | 230 | 221 | 247 | 252 | 314 | 277 |
| Trial 7 | 231 | 257 | 229 | 278 | 339 | 268 |
| Trial 8 | 298 | 275 | 217 | 275 | 224 | 277 |
| Trial 9 | 228 | 266 | 183 | 258 | 289 | 262 |
| Trial 10 | 246 | 285 | 251 | 305 | 197 | 261 |
| Average: | 250.6 | 277.2 | 247.1 | 278.1 | 277.1 | 263.3 |

Figure 1 – TCP/IP Direct and Relayed Echo simulation results

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **UDP** | | | | | | |
|  | **Relay Echo** | | | | | |
| Client | ohms | | laminar | | laminar | |
| Server | nano | | nano | | nano | |
| Destination | laminar | | ohms | | hertz | |
| Trial Type | User | ping | User | ping | User | ping |
| Trial Number | Time (microseconds) | | Time (microseconds) | | Time (microseconds) | |
| Trial 1 | 825 | 293 | 799 | 342 | 884 | 277 |
| Trial 2 | 821 | 313 | 889 | 336 | 913 | 333 |
| Trial 3 | 772 | 354 | 831 | 350 | 932 | 307 |
| Trial 4 | 779 | 308 | 725 | 329 | 961 | 336 |
| Trial 5 | 775 | 337 | 837 | 335 | 798 | 298 |
| Trial 6 | 873 | 346 | 735 | 328 | 786 | 321 |
| Trial 7 | 848 | 362 | 934 | 307 | 959 | 335 |
| Trial 8 | 818 | 309 | 855 | 345 | 820 | 326 |
| Trial 9 | 750 | 337 | 780 | 351 | 735 | 294 |
| Trial 10 | 831 | 306 | 814 | 328 | 835 | 237 |
| Average: | 825 | 326.5 | 819.9 | 335.1 | 862.3 | 306.4 |
|  | **Direct Echo** | | | | | |
| Server | nano | | nano | | nano | |
| Client | ohms | | laminar | | hertz | |
| Trial Type | User | ping | User | ping | User | ping |
| Trial Number | Time (microseconds) | | Time (microseconds) | | Time (microseconds) | |
| Trial 1 | 264 | 302 | 245 | 285 | 297 | 321 |
| Trial 2 | 280 | 260 | 341 | 333 | 289 | 314 |
| Trial 3 | 251 | 267 | 306 | 280 | 294 | 325 |
| Trial 4 | 283 | 257 | 279 | 283 | 298 | 315 |
| Trial 5 | 291 | 302 | 285 | 285 | 296 | 279 |
| Trial 6 | 293 | 275 | 246 | 275 | 345 | 296 |
| Trial 7 | 280 | 229 | 300 | 279 | 292 | 309 |
| Trial 8 | 301 | 250 | 377 | 281 | 299 | 310 |
| Trial 9 | 287 | 271 | 268 | 258 | 298 | 274 |
| Trial 10 | 319 | 238 | 294 | 243 | 268 | 301 |
| Average: | 284.9 | 265.1 | 294.1 | 280.2 | 297.6 | 304.4 |

Figure 2 – UDP Direct and Relayed Echo simulation results

After analysis of the collected data, we came to the conclusion that UDP is faster than TCP in the relayed echo, as expected. However, the TCP direct echo application had slightly faster successful transmissions than the UDP application. It’s not surprising that UDP is faster in sending packets to multiple destinations (as is the case in the delayed echo), as there is no error checking and thus faster transmission times. However, in the case of a single reception and transmission, as is the case of the direct echo, the two protocols have an insignificant difference in time. In reality, sending a single packet between a single server and client rarely happens.

Ping, however, is a software utility that measures the round-trip time for transmissions from a single client and server. Ping uses ICMP to transmit data, which while similar to UDP in that it doesn’t have acknowledgements, isn’t TCP or UDP. The Ping times were compared to both TCP and UDP direct echo tests, which isn’t surprising. s