Lab 1

CS 640

Learning objectives

1. Write applications that use sockets to transmit and receive data across a network
2. Describe how latency and throughput can be measured
3. Explain how latency and throughput are impacted by link characteristics and multiplexing

Things to Watch Out For

- Use Java 7 and Java Sockets
- Establish a TCP connection
- Work in a team of 2-3
- Follow all naming conventions
- Use correct units (Bytes in KB and rate in Mbps)
- Use proper documentation practices within code
- Write a README with names, usernames, and assumptions made
- Write a makefile to compile everything

Part 1: Write Iperfer

Client:

```
java Iperfer -c -h <server hostname> -p <server port> -t <time>
```

- -c : client mode
- -h <server hostname> : hostname or IP of server that should consume data
- -p <server port> : port of server that is waiting to consume data. 1024 <= p <= 65535
- -t <time> : duration of time in seconds data should be generated.

Part 1: Write Iperfer

Server

```
java Iperfer -s -p <listen port>
```

- -s : server mode
- -p <listen port> : port host is waiting to consume data on

- Only connects to 1 client and exits on EOF signal (closed connection from client)

Part 1: Write Iperfer

Error Messages

- If the error messages are invalid print and exit:

```
Error: invalid arguments
```

- If port number incorrect print and exit:

```
Error: port number must be in the range 1024 to 65535
```

- Create own validations for hostname and time that you think is reasonable.

Part 1: Write Iperfer

Steps after connection

- Client should send data (1000 bytes of 0x0) as quickly as possible. Keep running total
- Server should read data in 1000 byte chunks and keep running total of bytes received.
- Times up
- Client closes connection and prints:

```
sent=<# bytes sent> KB rate=<rate at which bytes were sent> Mbps
```

- Server closes on EOF signal from client and prints:

```
received=<# bytes received> KB rate=<rate at which bytes were received> Mbps
```

Part 1: Write Iperfer

Testing Wired

- Test on any machines you want that have a wired connection
- Watch out for firewall blocks
- Take screenshots of measurements for client and server
- Insert screenshots with appropriate labels in answers.pdf under heading **Part 1 - Wired Environment**

Part 1: Write Iperfer

Before Testing Wireless

Under which environment - wired or wireless, would you expect greater throughput?

Note your answer along with the reasoning in answers.pdf under the heading **Part 1 - Wireless Environment**.

Part 1: Write Iperfer

Testing Wireless

- Set up wireless connection. Potential options:
  - Connect two machines (laptop) to a wireless router. You can then run the server on one machine and the client on another machine.
  - Connect two machines to UW Net and do the experiment.
  - Set up a wireless hotspot on most desktops/laptops with some of the latest operating systems.
- Watch out for firewall blocks
- Take screenshots of measurements for client and server
- Insert screenshots with appropriate labels in answers.pdf under heading **Part 1 - Wireless Environment**

Part 1: Write Iperfer

After Testing Wireless

Did it match the prediction you made at the start of this section? In either case, explain your results.

Put your answers into answers.pdf under **Part 1 - Iperfer Results**.

Part 2: Mininet Tutorial

Use your personal machines:

- Follow instructions in description and at http://mininet.org/download/ and http://mininet.org/vm-setup-notes/
- Download VM such as VirtualBox
- Download Mininet
- Install Java on mininet
- Install X11
    - I used XQuartz on my Mac to accomplish this

Part 2: Mininet Tutorial

Follow: http://mininet.org/walkthrough/

- All of Part 1, except the section "Start Wireshark"
- The first four sections of Part 2—"Run a Regression Test", "Changing Topology Size and Type", "Link variations", and "Adjustable Verbosity"
- All of Part 3

Part 3: Measurements in Mininet

Q1 Link latency and throughput:

- Set up topology (described on page 6)
- Measure latency and measured throughput of each link L1-L5
  - To measure the latency / bandwidth between links, use the host endpoints (instead of switches). Open xterm for each host machine and run the required commands.
  - Run ping with 30 packets and store measurement output in file latency_L#.txt
  - Run Iperfer for 30 seconds and store measurement output in file throughput_L#.txt

Part 3: Measurements in Mininet

Q2 Path latency and throughput:

- Assume h1 wants to talk to h4. Explain expected latency and throughput on path in answers.pdf under heading **Part3 - Q2 Predictions**.
- Measure latency and measured throughput between h1 and h4
  - Run ping with 30 packets and store measurement output in file latency_Q2.txt
  - Run Iperfer for 30 seconds and store measurement output in file throughput_Q2.txt. which one is the server or client doesn't matter.
- Find the average and explain the result in answers.pdf under heading **Part 3 - Q2 Results**. Explain why they were wrong if they were.

Part 3: Measurements in Mininet

Q3 Effects of Multiplexing:

- Assume multiple hosts connected to s1 want to simultaneously talk to hosts connected to s4.
- Explain the expected latency and throughput when two pairs of hosts are communicating simultaneously? What about 3 pairs? Put predictions in answers.pdf under **Part3 - Q3 Predictions**
- Choose 2 pairs of hosts, one connected to S1 and thee second connected to S4.
    - Run ping with 30 packets and put the average RTT and measured throughput for each pair in answers.pdf under heading **Part 3 - Q3 Results**.
    - Run Iperfer for 30 seconds and put the average RTT and measured throughput for each pair in answers.pdf under heading **Part 3 - Q3 Results**.
- Is prediction was wrong, explain why in results section.
- Repeat for three pairs of hosts communicating simultaneously

Part 3: Measurements in Mininet

Q4 Effects of Latency:

- Assume h1 wants to communicate with h4 at the same time h5 wants to communicate with h6
- Explain the expected latency and throughput for each pair. Put predictions in answers.pdf under **Part3 - Q4 Predictions**
- Measure latency and measured throughput for h1-h4 and h5-h6
  - Run ping with 30 packets and store measurement output in files latency_h1-h4.txt and latency_h5-h6.txt
  - Run Iperfer for 30 seconds and store measurement output in files throughput_h1-h4.txt, and throughput h5-h6.txt
- Put average RTT and measured throughput in answers.pdf under **Part3 - Q4 Results** and explain. If you predictions were wrong, why?

Submission

To submit this lab, we are asking for 1 tar file per group with the CS username of each group member. (for exact naming convention and how to create a tar file please refer to the lab1 description page 8.) In each tar file there should be 3 things:

1. The source code for Iperfer—all Java source files for Iperfer should be in a folder called iperfer; the folder should include a Makefile that compiles the Java source.
2. Your measurement results and answers to the questions from Part 3—all measurements captured in text files and answers to questions in the file answers.pdf should be in a folder called measurement
3. A README file with the names and CS usernames of both group members

Upload the tar file on the Lab1 tab on course's Canvas page.

Late Policy

Upto 24 hours late — lose 10% of points
Upto 48 hours late — lose 30% of points
Upto 72 hours late — lose 60% of points
Beyond 72 hours — lose 100% of points

Rubric

Section 1: Prerequisites (5 points)

1 - correct submission format
2 - Code compiles
1 - code is properly structured
    0.5 - Code has useful comments
    0.5 - Code has proper indentation
1 - Command takes valid inputs

Rubric

Section 2: Client (10 points)

2 - args parsed correctly
    1 - Error shown on missing/additional/incorrect args
    1 - Error shown on illegal port number
4 - ClientSocket is correct, and works
    1 - Socket object
    1 - Connects to correct endpoint
    1 - Socket sends data
    0.5 - Each data chunk is 1000 bytes
    0.5 - data is initialized to 0x0
2 - data is seen for time period t (specified in command line args)
2 - Throughput calculation including display
    1.5 - throughput is calculated correctly with correct units
    0.5 - output format matches that in description

Rubric

Section 3: Server (10 points)

2 - args are passed correctly
    1 - Error shown on missing/additional/incorrect args
    1 - Error shown on illegal port number
4 - ServerSocket is correct, and works
    1 - ServerSocket object bound to correct port
    1 - Accepts connections
    1 - Received data
    0.5 - Tries to read 1000 bytes each time
    0.5 - count correct # of bytes received (not necessarily 1000)
2 - Time duration
    1 - Start time on accept or on arrival of 1st trunk of data
    1 - End time on client closing connection (receiving EOF)
2 - Throughput calculation including display
    1.5 - Rate calculated correctly with correct units
    0.5 - Output format matches that in description

Rubric

Section 4: Testing (6 points)

2 - Wired Environment
- 1 - Throughput values on client and server
- 1 - Screenshot

4 - Wireless Environment
- 1 - wrote a prediction
- 1 - Throughput values on client and server
- 1 - screen shot
- 1 - Explanation of results

Rubric

Section 5: Measurements in Mininet (29 points)

All measurements for 30 packets and 30 seconds (if incorrect deduction 1*num main question)

5 - Q1 Latency
- 1 - Latency_L1.txt
- 1 - Latency_L2.txt
- 1 - Latency_L3.txt
- 1 - Latency_L4.txt
- 1 - Latency_L5.txt

5 - Q1 Throughput
- 1 - Throughput_L1.txt
- 1 - Throughput_L2.txt
- 1 - Throughput_L3.txt
- 1 - Throughput_L4.txt
- 1 - Throughput_L5.txt

5 - Q2
- 2 - Prediction written
- 1 - Average RTT found
- 1 - Measured throughput found
- 1 - explanation of results

Rubric

Section 5: Measurements in Mininet (29 points)

7 - Q3
- 2 - Prediction written for 2 hosts
- 2 - Prediction written for 3 hosts
- 1 - Latency and throughput measurement for 2 hosts
- 1 - Latency and throughput measurement for 3 hosts
- 1 - Explanation of results

7 - Q4
- 2 - Prediction written
- 1 - Latency_h1-h4.txt
- 1 - Latency_h5-h6.txt
- 1 - Throughput_h1-h4.txt
- 1 - Throughput_h5-h6.txt
- 1 - Explanation of results