

Computer Networks

Lab 1

Yvan Petillot

Networks - Lab 0

1

What You Will Do In This Lab.

The purpose of this lab is to become familiar with tcp/ip and udp programming. You will experiment and develop code to put the theoretical knowledge we have seen into practice

You have several tasks before you:

- Using your favorite editor, edit the files to perform tcp/ip programming and udp programming to make it work
- You will know you are done when you have demonstrated to me that your program works.

Networks - Lab 0

2

Where To Get Documentation

Use the notes given in class (socket programming)

Use Beej's notes if you have printed them

Quick Unix reminder:

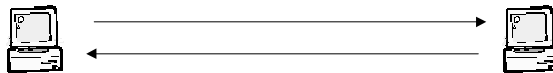
<u>Create a directory:</u>	mkdir
<u>Change directory:</u>	cd dir_name
<u>Move up one directory:</u>	cd ..
<u>Find out where you are:</u>	pwd
<u>List files in directory:</u>	ls
<u>Copy files:</u>	cp file location
<u>Move Files:</u>	mv file location
<u>Remove Files:</u>	rm file

Networks - Lab 0

3

Computer Chat

- How do we make computers talk?



- How are they interconnected?

Internet Protocol (IP)

Networks - Lab 0

4

Internet Protocol (IP)

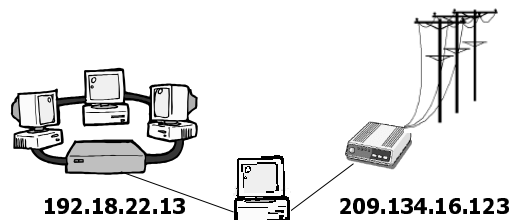
- **Datagram (packet) protocol**
- **Best-effort service**
 - Loss
 - Reordering
 - Duplication
 - Delay
- **Host-to-host delivery**

Networks - Lab 0

5

IP Address

- **32-bit identifier**
- **Dotted-quad: 134.111.10.43**
- **www.clarku.edu -> 140.232.1.19**
- **Identifies a host interface (not a host)**



Networks - Lab 0

6

Transport Protocols

Best-effort not sufficient!

- **Add services on top of IP**
- **User Datagram Protocol (UDP)**
 - Data checksum
 - Best-effort
- **Transmission Control Protocol (TCP)**
 - Data checksum
 - Reliable byte-stream delivery
 - Flow and congestion control

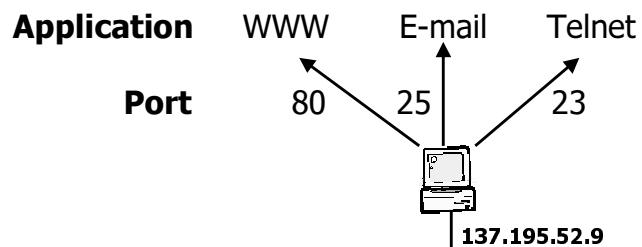
Networks - Lab 0

7

Ports

Identifying the ultimate destination

- **IP addresses identify hosts**
- **Host has many applications**
- **Ports (16-bit identifier)**



Networks - Lab 0

8

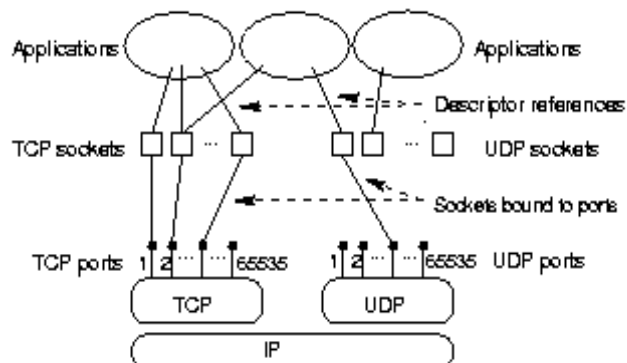
Socket

How does one speak TCP/IP?

- Sockets provides interface to TCP/IP
- Generic interface for many protocols

Sockets

- Identified by protocol and local/remote address/port
- Applications may refer to many sockets
- Sockets accessed by many applications



TCP/IP Sockets

- `mySock = socket(family, type, protocol);`
- TCP/IP-specific sockets

	Family	Type	Protocol
TCP	PF_INET	SOCK_STREAM	IPPROTO_TCP
UDP		SOCK_DGRAM	IPPROTO_UDP

- **Socket reference**
 - File (socket) descriptor in UNIX
 - Socket handle in WinSock

Networks - Lab 0

11

Specifying Addresses

```

Generic
• struct sockaddr
  {
    unsigned short sa_family; /* Address family (e.g., AF_INET) */
    char sa_data[14];        /* Protocol-specific address information */
  };

IP Specific
• struct sockaddr_in
  {
    unsigned short sin_family; /* Internet protocol (AF_INET) */
    unsigned short sin_port;   /* Port (16-bits) */
    struct in_addr sin_addr;   /* Internet address (32-bits) */
    char sin_zero[8];         /* Not used */
  };
  struct in_addr
  {
    unsigned long s_addr;     /* Internet address (32-bits) */
  };

```

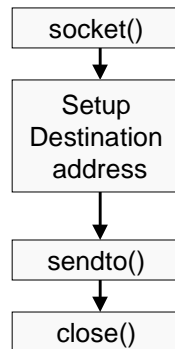
Networks - Lab 0

12

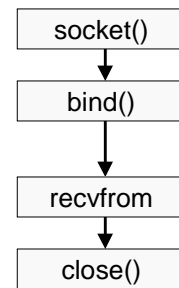
Overview of The Connection Mechanism

No Connection establishment
Connection less solution

UDP Talker



UDP Listener



Networks - Lab 0

13

What You Will Do In This Lab.

Go to

<http://www.ece.eps.hw.ac.uk/~ceeyrp/home/Pages/Teaching/B34LA1/B34LA1.html>

Download:

- udpListen.c
- udpTalk.c
- udpChat.c
- makefile
- tcp_server.c
- tcp_client.c

Task 1:

- Compile `tcp_server` and `tcp_client`
- Try running the server and several clients on the same machine
- Try running the server and several clients on multiple machines.
 - Remember to use `rlogin` to connect to remote machine (`rlogin machine name`).

Networks - Lab 0

14

What You Will Do In This Lab.

Task 2:

- Edit `udp_Listen` and `udpTalk` using emacs
- Check the code and identify the main elements of the protocol
- Correct errors introduced by your incompetent lecturer
- Compile and run the code
- Remove unnecessary code
- Change code to allow bi-directional communications

Task3:

- Download `udpChat`
- Correct errors introduced by your incompetent lecturer
- Compile and run the code
- Try to solve the problem of reception of your own messages

Task4:

- Make sure you read your notes again to consolidate what you have learnt.