CSE/EE 461: Lecture Topics

Neil Spring

September 17, 2000

This list outlines the topics to be covered in each lecture, and chapters in the book to read for background. As usual, this is subject to change. This is also your midterm and final review guide, since it lists each of the topics important that should be covered. Naturally, if we fall behind this somewhat ambitious schedule, you won't be responsible for material not covered in lecture. Some hints for using this list as a study guide are appended.

hw1..hw4 represent due dates for homeworks 1-4; pa1..pa3 represent due dates for programming assignments; midterm and final dates are also noted (they are in-class tests).

9/25 administrative, vocabulary, (1.2.2) 7-layer OSI model, (2.1) wire properties, baseline wander, clock recovery.

9/27 (2.2) encoding: NRZ, NRZI, manchester, 4B/5B, scrambling; overhead

9/29 (2.3.2) HDLC framing, bit stuffing; (2.4) error detection: (2.4.1) parity, (2.4.2) checksum, (2.4.3) CRC,

10/2 (1.3-1.3.2) Basics of Network Programming: bind, gethostbyname, connect, recv, sendto, select. packets as structs. Programming Assignment 1 assigned.

10/4 error correcting codes: hamming, 2 dimensional parity (3.4-3.4.4) buffering and switch fabrics: shared bus, crossbar, banyan; head of line blocking

10/6 (hw1) (3.1) hierarchical and flat addressing, forwarding tables and reachability.

10/9 (2.8) multi-access networks: RF, Aloha, slotted and (2.7.2) token ring,

10/11 (2.6-2.6.3) CSMA/CD, Ethernet: exponential backoff, 1-persistence, jamming, inter-frame spacing, minimum frame size. MTU. (2.8) CSMA/CA and 802.11 wireless

10/13(pa1) (4.1) IP network layer: (4.1.3) addressing, (4.1.2) service model, (4.1.4) packet switching, (4.1.7) control message protocol, (4.1.5) ARP

10/16 (3.2) repeaters, learning bridges and spanning tree, Routing.

10/18 (4.2) Routing: distance vector routing (Ford-Fulkerson, Bellman-Ford), (4.2.2, RFC 1058) RIP, counting to infinity, split horizon, poisonous reverse, triggered updates, fragmentation

10/20(hw2) Implementing a distance vector routing protocol: Programming Assignment 2 assigned.

10/23 Routing: (4.2.3) Link state, OSPF, (4.2.4) metrics, flooding, link-state packets and database, lollipop sequence space, dijkstra's algorithm

10/25 (not covered on midterm) BGP terminology, interior vs. exterior gateway protocols, (4.3.2) CIDR

10/27 (midterm) Midterm.

10/30 (2.5) Transport. Reliability, ordering, acknowledgement, timeouts, sequence numbers, (2.5.1) stop and wait / alternating bit protocol,

11/1 (2.5.2) sliding window, (5.2.4) flow control, waterfall diagrams, (5.2.5) loss detection by timeout and duplicate ack, SACK, NACK, (5.4) bandwidth delay product

11/3 (5.2.7) Silly window syndrome and avoidance (3 solutions), window scaling, 'Two Generals', (5.2.3) TCP state machine, connection setup, teardown,

11/6 tcpdump, sequence number plots

11/8 Implementing a sliding window transport protocol: Programming Assignment 3 assigned.

11/10(hw3) (6.3) congestion collapse, congestion control, AIMD, cwnd, ssthresh, slow start, congestion avoidance, ack pacing,

11/13 (5.2.5 ?) Timeouts, Karn's algorithm and RTT estimation, (6.4.3) TCP Vegas,

11/15 (6.4.2) RED, (6.4.1, sort of) ECN, delayed acks, Nagle's algorithm, Fragmentation and MTU discovery

11/17(pa2) Overlay networks, (4.4.2 inset) MBone, 6Bone, (4.4.3) dense / sparse (4.4) Multicast routing, ack implosion

11/20 flood / prune IGMP. (4.4.2) Reverse path forwarding, Scalable Reliable Multicast, Application Layer Framing, global ordering

11/22 (8.2) Security: principal, password authentication, salt. Attacks: DoS and amplification, sniffing, spoofing and Man in the middle, replay; (8.1) shared / symmetric key encryption,

11/27 (8.1.3) (RSA) public-private key encryption, (8.1.4) digital signatures, (8.3.2) key signing, web of trust (transitive trust),

11/29 Logic of authentication: (8.2.1: simple 3-way) challenge-response, freshness,

12/1(hw4) (8.2.1: Trusted third party) Kerberos: AS, TGT,, salt. server authentication

12/4 Telephone network: addressing, service model, (2.3.3) Time-division Multiplexing, circuit switching, SONET/DS/OC hierarchy

12/6(pa3) final review. end-to-end argument, stuff.

12/13 (final) Final exam, 12/13, 2:30-4:20.

When using this list as a study guide, for each term, ask yourself:

1. what is it?
2. what is it good for or why is it a problem?
3. what layer(s) (physical, data-link, network, transport, or application) does it belong to?
4. how does it work? you should be able to walk through an algorithm.

For example, *Time-division multiplexing* (10/13 lecture, and called *Synchronous Time Division Multiplexing* in the book) is a technique for squeezing regular data from several connections over the same wire. It's particularly good for the telephone network, where each phone call has its own time slice to transfer voice samples across the wire. It's a data-link layer technique, because it doesn't involve physical signals, and doesn't pass from link to link in a recognizable way. To implement TDM, each phone call is allocated a small slice of time, and a mapping is set up from incoming time slice number and port to outgoing time slice number and port at each switch. This is associated with circuit (as opposed to packet) switching.