The Internet Teaching Lab and Courses at UMass Amherst

Brian Neil Levine
Department of Computer Science
University of Massachusetts, Amherst

UMass Labs

- We have two labs, each in a separate room.
- Equipment is thanks to

- The CAIDA ITL equipment grant
  (1 of 3 cisco 7100 Routers)

- a 3-year NSF Combined Research-Curriculum Development (CRCD) grant (buys 13-20 PCs a year, plus pays for part-time tech person)

Courses

- There were two courses taught last Spring using ITL components.
  - *Introduction to Computer & Network Security* (Brian Levine)
  - *Multimedia Systems* (Prashant Shenoy)
- In the future:
  - Fall 01: Graduate *Computer Networking* (Levine)
  - Fall 01: Networking Lab course (Jim Kurose)
  - And the above courses again in Spring 2002.
- Eventually we want a on-going, "self-taught" lab-oriented course.

Security Class Objectives

An introduction to concepts in

- Cryptography
- Computer Security & Network Security
- supported with Practical experience with the systems and tools involved.

- Class consisted of 36 students (29 undergrads).
- The class was designed to be practical and discussion oriented.
- Jake Cunningham and Chris Misra, who are in charge of UMass computer and network security, also lectured and helped design the course.

Class Details

- We started with cryptography and 3 traditional homework assignments.
- The remainder of the course was based on 6 lab assignments
- Students also had to give one 5 minute presentation about that weeks Bugtraq news. (Really useful)

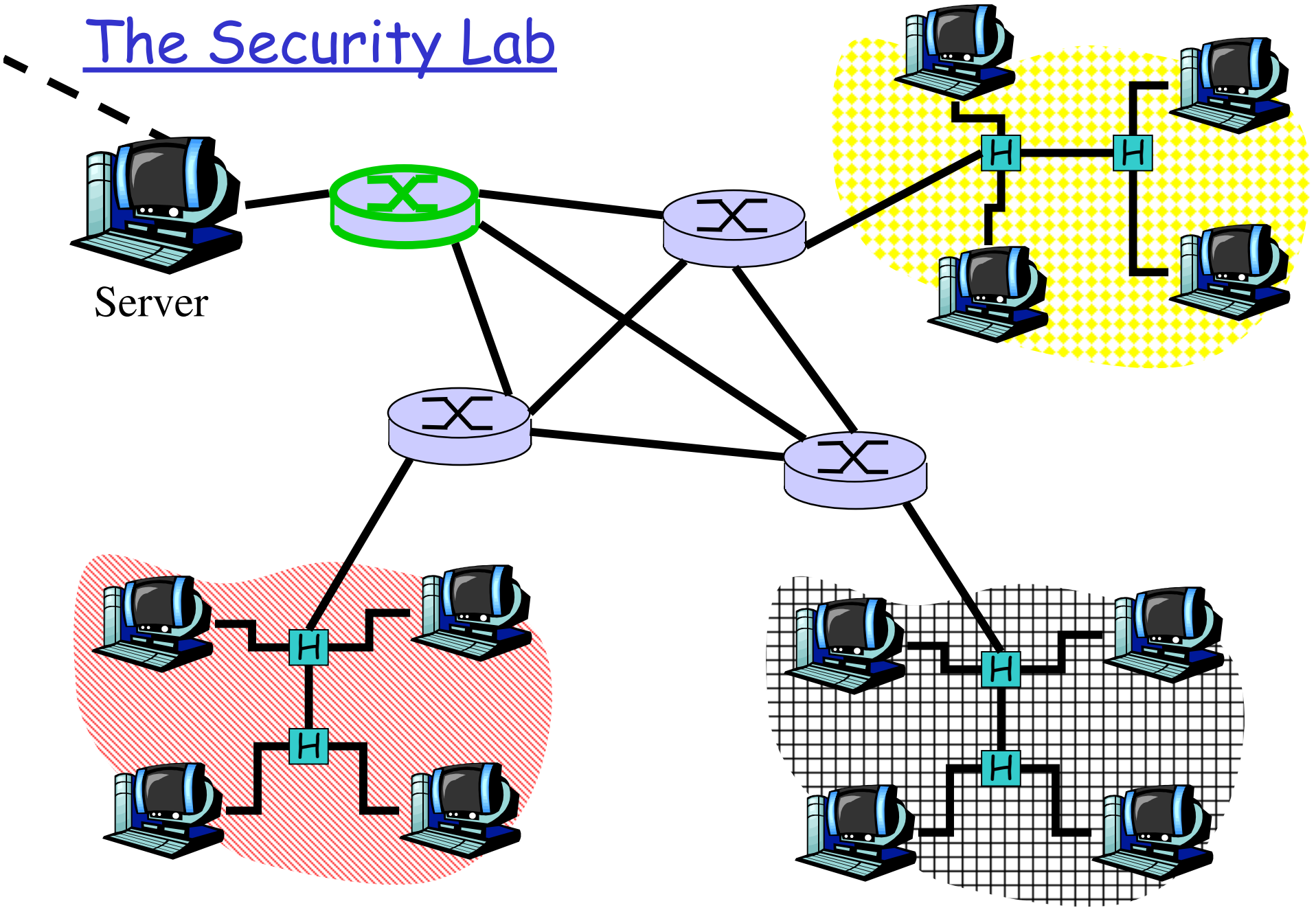
Course Topics

- Security Ethics
- Cryptography:
  - Block ciphers, (DES, AES, Blowfish), Public-key cryptography (RSA) and relevant number theory.
  - Hashes, key exchange, authentication protocols, Kerberos.
- Vulnerabilities and exposures, threat assesment.
- Securing your unix system (patching, unused services, tcp wrappers, etc).
- Buffer Overflow
- Sniffing: hacking versus legitimate uses. tcpdump, desniff/ssh, snort.

Course Topics (cont'd)

- Defending against Arp attacks, TCP session stealing and other problems with TCP/IP.
- Firewalling, DNS exposures, cache poisoning, and defenses.
- Denial of service, ddos.
- SSL, Cert. Authorties, virtual private networking (VPNs)
- Root kits, trojan horses, viruses, worms,
- Incident handling and recovery
- Anonymous Protocols and Privacy
- Intrusion Detection

The Security Lab



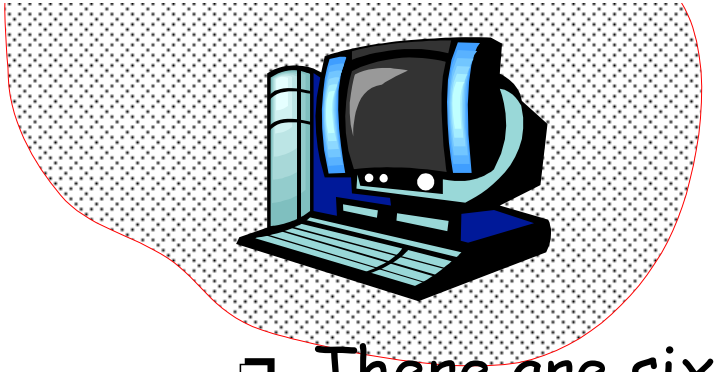
6 labs assignments

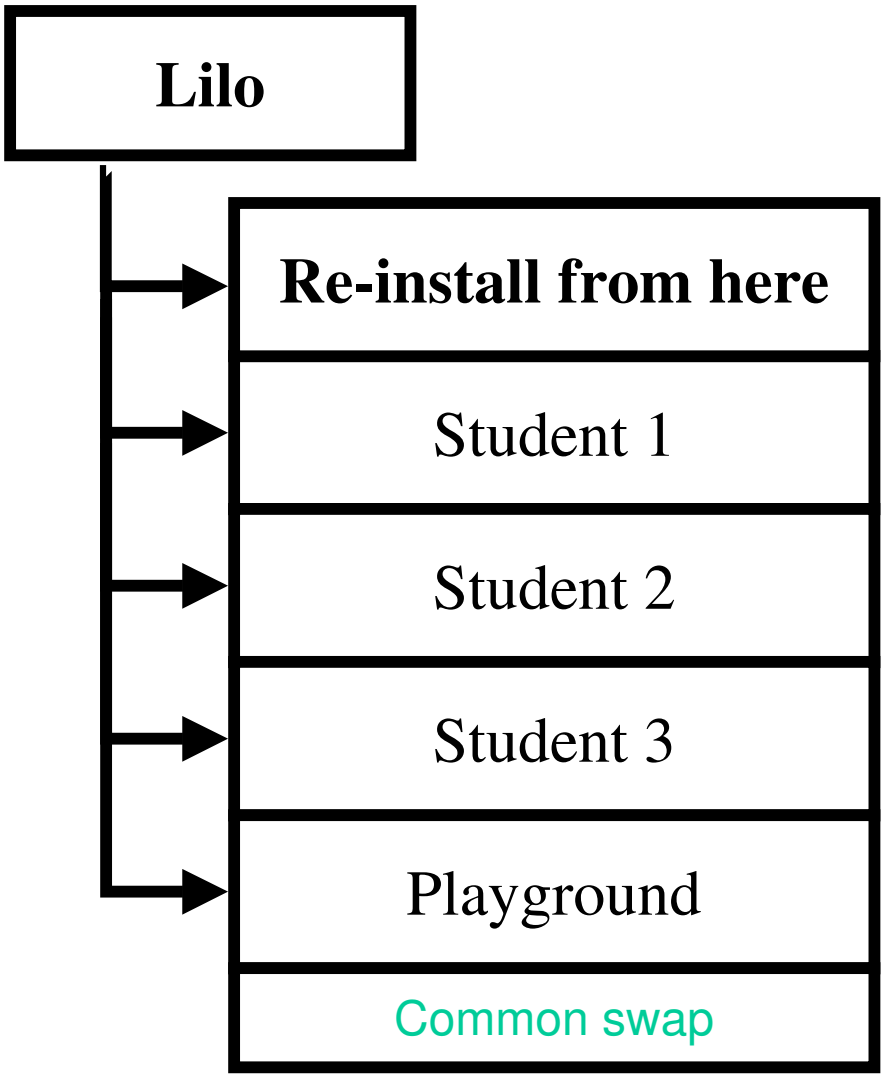
- Buffer overflow exploits
  - followed Phrack 49 for writing and running a exploit.
- Securing a linux workstation
  - ip-chains, turning off unused services, login restrictions, etc.
- Securing DNS
  - Configured "split" DNS, outside queries are treated differently than inside requests.
- Distributed Denial of Service Attacks
  - Ran and observered attacks
- Session Hijacking and Defenses
  - Observered TCP session hijacking and defenses (SSH)
- Using Snort for analyzing packet traces
  - Gave an unknown packet trace and students wrote snort monitoring rules to isolate packets.

Example Lab: Session Hijacking

- Students used Snort (or TCPdump) to log packets from a telnet connection from one machine to a remote machine.
- Next, we hijacked the session using a blind-spoofing attack implemtation.
- Students could observe the resulting ack storm and attack packets.
- Then, the same attack was attempted on an SSH connection.
- (It works, but fails to write acceptable data.)

Each machine





- There are six partitions on each machine
  - One password-protected partition for each student
  - One partition that anyone can use and over-write (a common class password)
  - One partition used to use while re-installing
  - (Swap space)

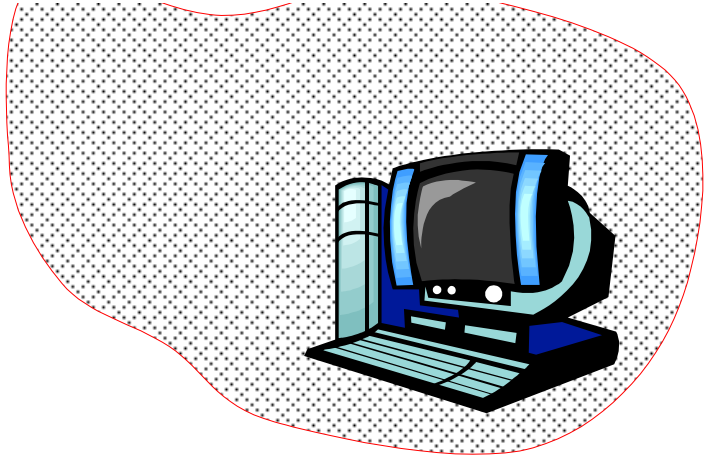
Practical Lessons Learned

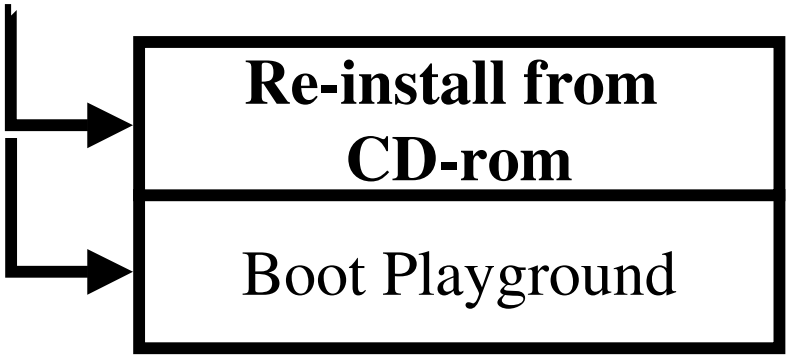
- We thought students would want their own partition.
- We though students would want the ability to save work on the server.
- We thought students would be experienced enough to know not to start assignments the night before.
- We thought we would have different installs for each lab.

- Students *loved* the practical part of the course.
- Organizing the lab exercises to work perfectly was challenging.

Lessons learned.

- It turns out having each machine be completely erasable is more flexible. When the lab was busy, students ended up just using the playground partition on arbitrary computers.
- Most lab work could be saved on a floppy.
- Next year, we plan to use staggered deadlines in some fashion, and labs that take about 3 hours and don't use more than 2 computers.
- It's simpler have each lab work off a single install.
- 12 computers seemed enough for 35 people, but tight.



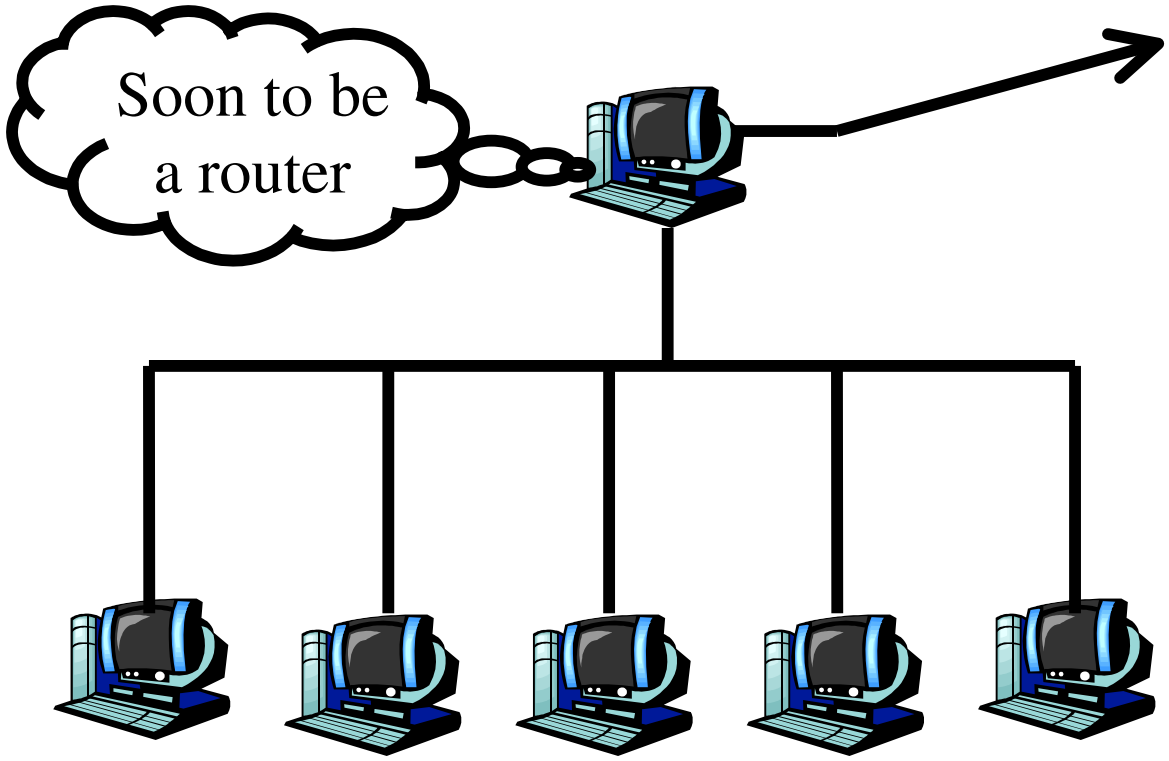


Next year...

- We are going to tape a CD-rom to the wall.
- One partition that anyone can use and over-write (a common class password)
- Students save work to floppies.

Multimedia Teaching Lab test bed

- 5 macines on a private network.
- Server with outside network access.
- Flexibility in configured network topology.



Sample Students Projects

- Implemented "lazy receiver" processing in the kernel
- Implemeneted a new scheduling algorithm in the kernel.
- Experiments with linux as a software router.
- Parallelized the mpeg-2 decoder
- Studies of multimedia middleware (RT-Corba)

Summary

- Setting up a practical curiculum was challenging
- but students found it invaluable
- and it was very exciting to do as a teacher!
- Labs really need to be ironed out well, and the lab set up has to be well thought out.
- We expect next year's offerings of the same courses to be smooth sailing and so we expect to try more crazy ideas.

- Eventually, we want a lab binder full of tens of lab exercises, and a course where students must complete some self-chosen subset.