Announcements

- Project #6 is available
- Reading Chapter 14 (Distributed Systems)
Monitoring

- **Record (log) significant events**
  - attempts to login to the system
  - changes to selected files or directories

- **Possible to compromise the log**
  - the user or software breaking in could delete all or part of the logs
  - could record logs to non-erasable storage
    - have a line printer attached to the machine
    - use WORM drives
  - send data to a secure remote host
Tripwire

- **Compute a set of expectorations about system**
  - Hash of file contents
  - Dates on files
- **Store database of values**
  - On read-only media
  - Offline
- **Periodically**
  - Compare database to current system
  - Report any differences
Encryption: protecting info from being read

- **Given a message** \( m \)
  - use a key \( k \), and function \( E_k \) to compute \( E_k(m) \)
  - store or send only \( E_k(m) \)
  - use a second second key \( k' \) and function \( D_{k'} \) such that
    - \( D_{k'}(E_k(m)) = m \)
    - \( E_k \) and \( D_{k'} \) need not be kept a secret

- **If** \( k = k' \) it’s called **private key encryption**
  - need to keep \( k \) secret
  - example DES

- **if** \( k \neq k' \), it’s called **public key encryption**
  - need only keep one of them secret
  - if \( k' \) is secret, anyone can send a private message
  - if \( k \) is secret, it is possible to “sign” a message
  - still need a way to authenticate \( k \) or \( k' \) for a user
  - example RSA
Public Key Encryption

- **Split into public and private keys**
  - public key used to encrypt messages
    - publish this key widely
  - private key used to decrypt messages
    - keep this key a secret

- **RSA**
  - algorithm for computing public/private key pairs
  - based on problems involved in factoring large primes
  - for an n bit message P, $C = (P^e \mod n)$, and $P = (C^d \mod n)$

- **Other Public Key Algorithms**
  - knapsack
    - given a large collection of objects with different weights
    - public key is the total weight of a subset of the objects
    - private key is the list of objects
Transposition Cipher

- **To Break:**
  - each letter is itself, so normal distribution of letters is seen
  - guess number of columns (verify with known plaintext)
  - order columns using trigram frequency
- **Block of text is used to break up digrams**

From: *Computer Networks*, 3rd Ed. by Andrew S. Tanenbaum, (c)1996 Prentice Hall.
DES

- Block cipher: uses 56 bit keys, 64 bits of data
- Uses 16 stages of substitution
- Variations
  - cipher block chaining: xor output of block n with into block n+1
  - cipher feedback mode: use 64bit shift register
    - can produce one byte at a time

From: Computer Networks, 3rd Ed. by Andrew S. Tanenbaum, (c)1996 Prentice Hall.
One Time Pad

- **Key Idea:** randomness in key
- **Create a random string as long as the message**
  - each party has the pad
  - xor each bit of the message with the a bit of the key
- **Almost impossible to break**
- **Some practical problems**
  - need to ensure key is not captured
  - a one bit drop will corrupt the rest of the message
Secure Socket Layer

- **Goal:**
  - Provide secure access to remote services
  - Authenticate remote servers to local users
  - Allow remote systems to authenticate users
  - Permit encrypted communication

- **Approach**
  - Public Key Cryptography
    - Certificates (signed by certificate authorities)
  - Server sends:
    - Certificate (signed use CA’s private key)
    - Certificate contains server’s public key
    - Client responds by encrypting reply using servers public key
    - Server checks response with private key
Sending Data

- **Data is split into packets**
  - limited size units of sending information
  - can be
    - fixed sized (ATM)
    - variable size (Ethernet)

- **Need to provide a destination for the packet**
  - need to identify two levels of information
    - machine to send data to
    - comm abstraction (e.g. process) to get data
  - address may be:
    - a globally unique destination
      - for example every host has a unique id
    - may unique between hops
      - unique id between two switches
Ethernet

- 10 Mbps (to 100 Mbps)
- milisecond latency
- limited to several kilometers in distance
- variable sized units of transmission
- bus based protocol
  - requests to use the network can collide
- addresses are 48 bits
  - unique to each interface
Hub based Ethernet

- Logically it is still a bus
- Physically, it is a star configuration
  - the hub is at the center of the network
- Hubs provide:
  - better control of hosts
    - possible to restrict traffic to only the desired target
    - can shutdown a host’s connection at the hub if its Ethernet device is misbehaving
  - easier wiring
    - can use normal telephone wire to connect links (called 10 base-T)
- 100 Megabit Ethernet
  - is only available with Hubs
  - requires different hubs than 10base-T
Ethernet Collisions

- If one host is sending, other hosts must wait
  - called Carrier Sense with Multiple Access (CSMA)
- Possible for two hosts to try to send at once
  - each host can detect this event (cd- Collision Detection)
  - both hosts must re-send information
    - if they both try immediately, will collide again
    - instead each waits a random interval then tries again
- Only provides statistical guarantee of transmission
  - however, the probability of success if higher than the probability of hardware failures and other events
ATM (Asynchronous Transfer Mode)

- 155Mbps and up
- fixed sized unit of transmission called a cell
  - cells are 48 bytes plus 5 bytes header
- switch based protocol
- for both local area and wide area networking
- addresses are VCI
  - virtual circuit ids
TCP/IP Protocol

- Name for a family of Network and Transport layers
  - can run over many link layers:
    - Arpanet, Ethernet, Token Ring, SLIP/PPP, T1/T3, etc.

- IP - Internet Protocol
  - network level packet oriented protocol
  - 32 bit host addresses (dotted quad 128.8.128.84)
  - 8 bit protocol field (e.g. TCP, UDP, ICMP)

- TCP - Transmission Control Protocol
  - transport protocol
  - end-to-end reliable byte streams
  - provides ports for application specific end-points

- UDP - user datagram protocol
  - transport protocol
  - unreliable packet service
  - provides ports for application specific end-points
TCP/IP History

- **Arpanet was the origin of today’s Internet**
  - started in 1969 to connect universities and DoD sites
  - early example of packet switched network
  - original links were 64kbps and 9.6kbps

- **Current TCP protocol**
  - started in use Jan 1, 1983
  - This was a *flag day*
    - all systems had to change to the new protocol at once
    - with the modern Internet this would be **hard** to do
Subnet Addressing

- Single site which has many physical networks
  - Only local routers know about all the physical nets
  - Site chooses part of address that distinguishes between physical networks
- subnet mask: splits the IP address into two parts
- Common Class B site mask 255.255.255.0
  - use 3rd byte to represent physical net
  - use 4th byte to represent host

vanilla scheme

subnet scheme
Encapsulation

How do we send higher layer packets over lower layers?

- **Higher level info is opaque to lower layers**
  - it’s just data to be moved from one point to another

  IP Header | IP Data Area

- **Higher levels may support larger sizes than lower**
  - could need to *fragment* a higher level packet
    - split into several lower level packets
    - need to re-assemble at the end
  - examples:
    - ATM cells are 48 bytes, but IP packets can be 64K
    - IP packets are 64K, but files are megabytes