Announcements

• Reminders:
  – Project 4 is “due” today
  – HW 6 is due today

• Final Exam:
  – Final Exam. Tuesday, Dec. 21st
  – 11:00-1:00
  – Towne 311
Current Grade Distribution

*Rough estimate*

Does not include:
- HW6 (3%)
- Project 4 (10%)
- Final (15%)

Current (normalized):
Average = 79%
Median = 79%
Std. Dev. = 5.7

(normalized)
< 65 = D
65-72 = C
72-79 = B
80 > = A

average
median
D
B
C
A
Today

• Course review

• Questions

• Course Evaluation
Main Take-away Ideas (1)

- Principles of Secure System Design
- Security is a process
- Least privileges
- Complete Mediation
- System Design
  - Economy of mechanism
  - Open standards
  - Failsafe Defaults
Main Take-away Ideas (2)

• Cryptography is important…
  – Can be used for more than just hiding information
  – Authentication and integrity

• … but not the only facet of security
  – Other risks
  – Social engineering is effective
  – Cryptography applied inappropriately is useless

• So: use it where necessary, and use it correctly
  – See Schneier’s book *Applied Cryptography*
Main Take-away Ideas (3)

• Concepts of security:
  – Confidentiality
  – Integrity
  – Availability

• General Mechanisms
  – Authentication
    • Challenge / Response
  – Authorization
    • Access control matrices
  – Audit
    • Logs
Main Take-away Ideas (4)

- Cryptography & Protocol Design
  - Shared vs. Public key cryptography

- Cryptographic protocols can be used for:
  - Authentication, privacy, confidentiality

- Challenge—Response is the fundamental method of authentication

- Nonces, Time stamps, Sequence numbers prevent replay attacks
Main Take-away Ideas (5)

• Malicious Code
  – Viruses & Worms
  – Defense in depth: patching, firewalls, proper configuration, auditing

• Buffer overflows are the #1 vulnerability
  – Choose safe languages:
    • Java, C#, Scheme, ML
  – Be aware of format string and input errors, take care when writing programs and scripts.
  – Software audit and design is important.
  – If you must use C or C++, use StackGuard, ProPolice, or another buffer-overflow preventative measure.
Future Directions - Courses

• Type systems, static analysis of programs, understanding programming languages, stack layouts, program compilation, optimization
  – CSE 340: Principles of Programming Languages
  – CSE 341: Compilers and Interpreters

• Internet software design
  – CSE 455: Internet and Web Systems

• Cryptography
  – Algebra, Number Theory courses, Crypto courses offered in the math department (Math 340, Math 690)

• Security
  – CIS 551: Networks and Security

• Networks
  – TCOM 510: Wireless Networking
K_{AB}\{\text{Let’s close this session, Bart, } n_A, n_B}\}

K_{AB}\{\text{Bye, Alice, } n_A, n_B'}\}