Prolog to Lecture 14
CS 236
On-Line MS Program
Networks and Systems Security
Peter Reiher

Secure Software and the Law

- Almost all software comes with no guarantees
  - Much less security guarantees
- If it's hacked, too bad
  - No liability to software creator
- Similarly, little or no liability for running insecure systems

Should That Change?

- What if software companies had to warrantee their software?
  - With liability if it failed in certain ways
- That would change their behavior
- But would it make the world safer?

How Would We Do This?

- Has to be something companies can realistically do
  - Meet standards
  - Use approved tools
  - Stand up to specified testing
- Can't just be, "you're hacked, you pay"
- Rules must be clear and consistent

Standards Approaches

- We'll talk about security standards in a future lecture
- But, briefly, hard to set comprehensive standards
- Standards don't cover all problems
- Who sets the standards?

Use Approved Tools

- Languages, development environments, testing tools
  - E.g., "can't use C"
- Would likely get rid of lots of problems
  - But certainly not all
- Easy to write insecure code with good tools

Specified Testing Approaches

- Independent groups run tests of software
- In pre-specified ways
- Presumably, same tests used for everyone
  - What would those tests be?
- Doesn't take into account new attacks

And What About Open Source?

- Would liability apply to open source code?
- If so, who would ever release open source?
- Or even non-open hobbyist code?
- Generally, would this limit software development to big companies?

How Would Liability Work?

- Would it require lawsuits?
- What level of proof would be required?
- What would constitute damages, of what size?
- Could you ever successfully sue Microsoft or another big company?