# Java Security

Edward W. Felten

Princeton University

# Why Java?

- meets a need
  - users want to browse
  - dynamic, interesting pages
- contrast to ActiveX
  - Java risk: hostile code breaks sandbox
  - ActiveX risk: user trusts too many programs

# Java Security Basics

- complexity the root of problems
  - usual development pressures
- depends on type-safe language
- most breaches have been due to breakdowns in type safety
- denial of service not addressed
- overall, security has improved, but problems remain

# Language Soundness

- type safety depends on language semantics
  - semantic problems lead to security breaches
- need definition and proofs
  - strains the limits of formal methods
- some problems found already
  - dynamic linking attacks

# Future Issues

- remote invocation and persistent objects
- garbage collection and finalization
- flexible security mechanisms
- complexity of JIT compilation
- generally, new features harbor bugs