Outline

- Designing and Writing Secure Code
  - General principles for architects/managers
  - Example: sendmail vs qmail (optional in backup slides)
- Buffer Overflow Attacks
- Defense for Buffer Overflow Attacks

General Principles

- Compartmentalization
  - Principle of least privilege
  - Minimize trust relationships
- Defense in depth
  - Use more than one security mechanism
  - Secure the weakest link
  - Fail securely
- Promote privacy
- Keep it simple
- Consult experts
  - Don't build what you can easily borrow/steal
  - Open review is effective and informative

Have you applied them in your design / evaluation?

Compartmentalization

- Divide system into modules
  - Each module serves a specific purpose
  - Assign different access rights to different modules
    - Read/write access to files
    - Read user or network input
    - Execute privileged instructions (e.g., Unix root)
- Principle of least privilege
  - Give each module only the rights it needs
- Minimize trust relationships
  - Clients, servers should not trust each other
    - Both can get hacked
  - Trusted code should not call untrusted code