Project 2 — Kernel Interception

Professor Hugh C. Lauer
CS-3013 — Operating Systems

(Slides include copyright materials from *Operating Systems: Three Easy Step*, by Remzi and Andrea Arpaci-Dusseau, from *Modern Operating Systems*, by Andrew S. Tanenbaum, 3rd edition, and from other sources)

Project 2

- **Assigned Friday, January 19**

- **Checkpoint Sunday, January 28**
  - I.e., nine days after assignment!

- **Due Friday, February 2**
  - I.e., two weeks from today!

Caution — Caution — Caution

- **You don’t know what it is that you don’t know**
  - I.e., what you need to learn in order to carry out the project!

Phase 1 — On-access anti-virus scanner

- **Intercept and modify existing system call(s)**
  - Open
  - Close
  - Read

- **Record opening, closing, and reading in system log**
  - But not for root or known system “users”

- **On reading, scan for string “VIRUS”**

Phase 1 (continued)

- **Implement with Loadable Kernel Module**
- **Insert in Project 0 kernel**
  - Replace `cs3013_syscall1`
  - No need to recompile kernel
    - `insmod`

Phase 2:– Process Genealogy

- **Find ancestors, children, and siblings of specified process.**
  - Use `cs3013_syscall2`

- **Use space program to work with kernel call**
  - `copy_to user()`
  - `copy_from_user()`

User-space test program

- **To test and demonstrate correctness of both parts**
  - Needed for grading and demos to TAs

Strongly encouraged to work in teams of two

Register partnership in InstructAssist

OR

Ask InstructAssist to pair you with a random partner.

Questions?