CS 419 Computer Security

Week 0: About the Class

Paul Krzyzanowski

Welcome to computer security

About Me

AT&T Bell Laboratories

↓

Lucent Technologies
Bell Labs Innovations

↓

Start-ups
- → NetDox
- → IDMetrix
- → Face2Face

↓

StorageApps
- → HP

↓

OPENPEAK
- → Verizon, AT&T
- → Cisco
- → BlackBerry

↓

Xevo By LEAR
- → Toyota, GM, Ford
- → Mazda, Honda, Subaru
- → Fiat Chrysler, …

Web site and contact

web: www.cs.rutgers.edu/~pxk/419

mirror: www.pk.org/419

email: pxk@cs.rutgers.edu

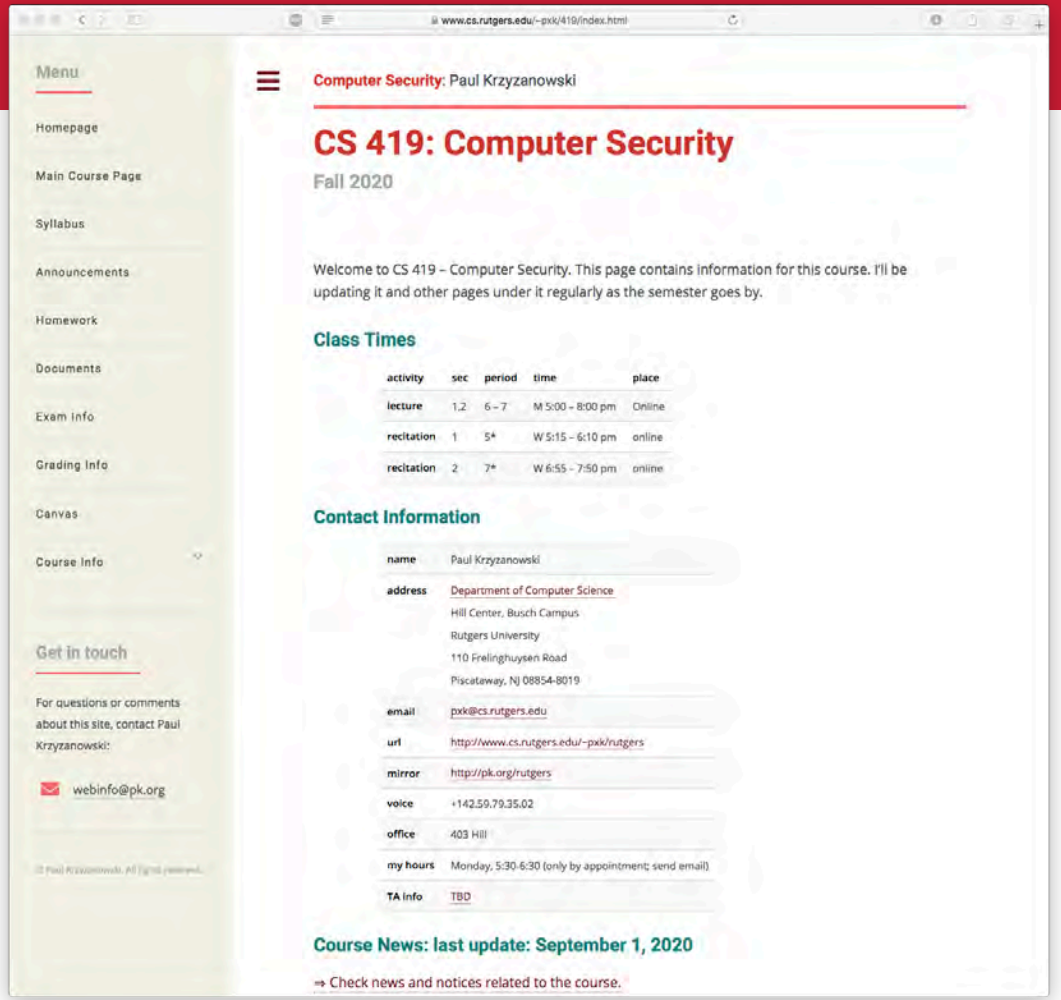
phone: +142.59.79.35.02

zoom: https://rutgers.zoom.us/my/pxk.rutgers

Canvas: https://rutgers.instructure.com/courses/66088

Class info

- Contact info
- TA info
- Syllabus
- Lecture notes
- Class news
- Homework



Class meetings

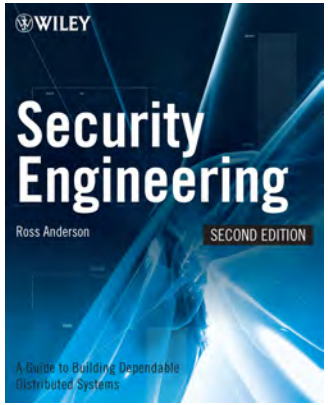
- **Classes will be held via Zoom**
- **Lecture recordings will be made available**
  - YouTube? Canvas?
  - Web site
- **You can take classes asynchronously**
  - I'll be around for questions
  - Send via zoom chat, email, etc.
  - I will post FAQs and corrections if needed

Text

None required – but we will use several sources

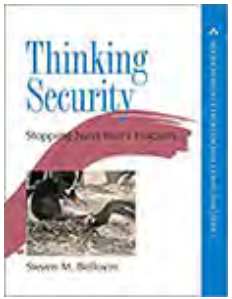
Security Engineering: A Guide to Building Dependable Distributed Systems
2nd Edition
by Ross J. Anderson

Free at
https://www.cl.cam.ac.uk/~rja14/book.html



But mostly…

- Other reading material on the web
- Lecture slides
- Lecture notes/summaries



Policy

- **Weekly quizzes: 5-6 questions; 15 minutes**
- **Short programming assignments (~5)**
  - Individual assignments
  - Due prior to the due date
- **Written assignments**
  - Due prior to the due date
  - No MS-Word, Pages, InDesign submissions!
  - PDF or text with line breaks only (HTML for in-line text on sakai)
- **Collaboration & academic integrity**
  - Individual assignments – no copying!

Grades

- **Quizzes** ~ 50%
- **4-5 written assignments** ~ 15%
- **~4-5 programming assignments** ~35%

What this course IS

- **Security engineering**
- **Understand why systems have weaknesses**
- **How do we deal with these weaknesses?**
  - People, devices, networks, operating systems, applications
  - Cryptographic algorithms
  - Authentication & key distribution protocols
  - Ensuring integrity & confidentiality

Things we'll cover

- **Intro: threats, risks, security needs**
- **Access control**
  - Core OS mechanisms for access contol
  - Mandatory vs. discretionary access control
- **Code injection**
  - Buffer overflow, shell scripts, input validation
- **Client-side risks & protection**
  - Viruses, worms, trojans
  - Human factors
- **App confinement**
  - Jails, virtual machines, sandboxes
- **Cryptography**
  - Encryption
- **Integrity & key distribution**
  - Public keys, hashing, digital signatures
- **Authentication**
  - Passwords, tokens, biometrics, cards
- **Cryptocurrency**
  - Bitcoin, proof of work, proof of stake
- **Network security**
  - Switches, routers, services
- **Network protection**
  - Firewalls, VPNs
- **Web security**
- **Mobile security**
- **Anonymous communication**
  - Tor
- **Content protection**
  - Steganography, watermarking, DRM
- **IoT**
  - Security risks in embedded software
- **Forensics**

What this course is NOT

- **How to write viruses/trojans/malware**
- **How to break into systems**
- **How to be a hacker**
- **Rigorous mathematics on systems, security, or cryptography**

The End