

# CS 458 / 658

## Computer Security and Privacy

Module 1  
Introduction to Computer Security and Privacy

Winter 2011

# Course mechanics

- Instructors:
- Section 1: TTh 10:00 am–11:20 pm, DWE 3517
  - Urs Hengartner
  - <http://www.cs.uwaterloo.ca/~uhengart/>
  - Office hours: Thursdays 1:30–2:30 pm or by appointment in DC 3526
- Section 2: TTh 8:30–9:50 am, MC 4042
  - Ian Goldberg
  - <http://www.cs.uwaterloo.ca/~iang/>
  - Office hours: Wednesdays 10–11 am or by appointment in DC 3518

# Course mechanics

- Teaching assistants: Ryan Henry, Mehrdad Nojournian, Rob Smits, Colleen Swanson
- **Come to class!** Not every bit of material will be on the slides or in the text.
- You will need an account in the student.cs environment
  - **If you don't have a student.cs account for some reason, get one set up in MC 3017.**

# Course website

- This course will use UW-ACE extensively
  - Syllabus, calendar, lecture notes, additional materials, assignments, discussion, announcements, policies, etc.
  - Site will be updated regularly
  - It is your responsibility to keep up with the information on that site.
- Feedback is encouraged!
  - Anonymous suggestion box in UW-ACE

# Additional communication

- Some communication might be sent to your UW email address
  - Check UW email account regularly or have email forwarded to your regular account
- Use discussion forums in UW-ACE for questions of general interest
- Use UW-ACE course mail for questions just for course personnel
- Use your regular UW email account if for some reason UW-ACE is not working

# Course syllabus

- You are expected to be familiar with the contents of the course syllabus
- Available in UW-ACE
- If you haven't read it, read it after this lecture

# Plagiarism and academic offenses

- We take academic offenses very seriously
  - Even (especially?) in fourth year
- Nice explanation of plagiarism online
  - [http://arts.uwaterloo.ca/arts/ugrad/academic\\_responsibility.html](http://arts.uwaterloo.ca/arts/ugrad/academic_responsibility.html)
- Read this and understand it
  - Ignorance is no excuse!
  - Questions should be brought to instructor
- Plagiarism applies to both text and code
- You are free (even encouraged) to exchange ideas, but **no sharing code or text**

# Plagiarism (2)

- Common mistakes
  - Excess collaboration with other students
    - Share ideas, but no design or code!
  - Using code from other sources (like previous offerings of this course)
- Possible penalties
  - First offense (for assignments; exams are harsher)
    - 0% for that assignment, -5% on final grade
  - Second offense
    - Expulsion is possible
- More information linked to from course syllabus



# Grading scheme

- Midterm (15%)
  - Tuesday, February 15, 7:00 pm
- Final (30%)
  - You must pass the weighted exam mark in order to pass the course!
- Assignments (45%)
  - Work alone
- Self-tests (5%)
- Blog task (5%)
- Additional research survey paper for CS 658
  - See syllabus in UW-ACE for more details
- See syllabus for late and reappraisal policies, academic integrity policy, and other details

# Assignments

- Assignments will be due **at 3 pm**
- Late submissions will be accepted up to 48 hours after due date
- There will be no penalty for accepted late submissions
- Multiple assignments can be submitted late, including the last one
- **No assistance will be given after the due date**
- You need to notify your instructor **before the due date** of a severe, long-lasting problem that prevents you from doing an assignment

# Self-tests

- The self-tests are worth 5% of your grade
- They're meant to help you keep up with the material, and gauge your grasp of it on an ongoing basis
- Check calendar in UW-ACE for the availability and deadline information for each self-test
  - First test: available tomorrow, deadline one week
  - **No late self-tests will be accepted!**
  - You can attempt each self-test as often as you like during its availability period; your last grade on each self-test will be the one recorded
- Format: online (on UW-ACE), usually multiple-choice questions

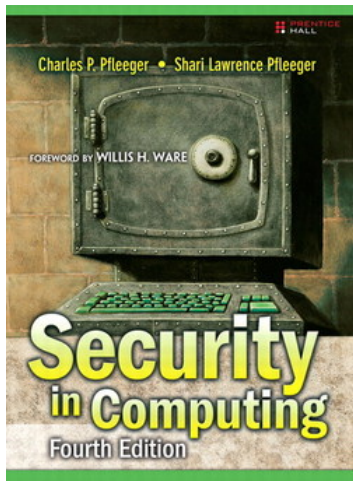
# Blog task

- Many of the security and privacy problems that we will discuss in this course will (unfortunately) occur in the real world during the next four months
- The blog task forces you to keep up with these developments
- Each student has to write one blog post during an assigned timeslot
- You must also participate in the discussion of other students' blog posts **throughout the term**
- **Blog task is part of material covered in exams**
- See UW-ACE for sign-up and other instructions

# A note on security

- In this course, you will be exposed to information about security problems and vulnerabilities with computing systems and networks
- To be clear, **you are not to use this or any other similar information to test the security of, break into, compromise, or otherwise attack, any system or network** without the express consent of the owner
- In particular, you will comply with all applicable laws and UW policies
- See syllabus in UW-ACE for more details

# Required textbook



- **Security in Computing**, 4th edition, Charles P. Pfleeger and Shari Lawrence Pfleeger, Prentice-Hall, 2007.
- You are expected to know
  - entire textbook sections, as listed on course website
  - all the material presented in class

<http://proquest.safaribooksonline.com/9780132390774>

# Other readings

- From time to time, there will be other readings assigned as well
- They will be linked to from the modules page in UW-ACE
- There will be both mandatory and optional readings
- You must read the mandatory ones **before** the class in which we will discuss them
  - There is such a reading for the next lecture

# Module outline

- ① What is our goal in this course?
- ② What is security?
- ③ What is privacy?
- ④ Who are the adversaries?
- ⑤ Assets, vulnerabilities, threats, attacks, and controls
- ⑥ Methods of defence



# What is our goal in this course?

- Our primary goal is to be able to **identify security and privacy issues** in various aspects of computing, including:
  - Programs
  - Operating systems
  - Networks
  - Internet applications
  - Databases
- Secondly, to be able to use this ability to **design systems that are more protective of security and privacy**.

# What is security?

- In the context of computers, **security** generally means three things:
  - **Confidentiality**
    - Access to systems or data is limited to authorized parties
  - **Integrity**
    - When you ask for data, you get the “right” data
  - **Availability**
    - The system or data is there when you want it
- A computing system is said to be secure if it has all three properties
  - Well, usually

# Security and reliability

- Security has a lot to do with reliability
- A secure system is one you can rely on to (for example):
  - Keep your personal data confidential
  - Allow only authorized access or modifications to resources
  - Give you correct and meaningful results
  - Give you correct and meaningful results **when you want them**

# What is privacy?

- There are many definitions of privacy
- A useful one: “informational self-determination”
  - This means that you get to control information about you
  - “Control” means many things:
    - Who gets to see it
    - Who gets to use it
    - What they can use it for
    - Who they can give it to
    - etc.

# Example: PIPEDA

- PIPEDA (Personal Information Protection and Electronic Documents Act) is Canada's private-sector privacy legislation
- Lists ten Fair Information Principles companies have to abide by:
  - Be accountable
  - Identify the purpose of data collection
  - Obtain consent
  - Limit collection
  - Limit use, disclosure and retention
  - Be accurate
  - Use appropriate safeguards
  - Be open
  - Give individuals access
  - Provide recourse

# Security vs. privacy

- Sometimes people place security and privacy as if they're opposing forces.
- Are they really? Do we have to give up one to get the other?

# Who are the adversaries?

- Who's trying to mess with us?
- Various groups:
  - Murphy
  - Amateurs
  - “Script kiddies”
  - Crackers
  - Organised crime
  - Government “cyberwarriors”
  - Terrorists
- Which of these is the most serious threat today?

# How secure should we make it?

- Principle of Easiest Penetration
  - “A system is only as strong as its weakest link”
  - The attacker will go after whatever part of the system is easiest for him, not most convenient for you.
  - In order to build secure systems, we need to **learn how to think like an attacker!**
  - How would you get private information from the US Social Security Administration database?
- Principle of Adequate Protection
  - “Security is economics”
  - Don't spend \$100,000 to protect a system that can only cause \$1000 in damage



# Weakest link



# Some terminology

- **Assets**
  - Things we might want to protect, such as:
    - Hardware
    - Software
    - Data
- **Vulnerabilities**
  - Weaknesses in a system that may be able to be **exploited** in order to cause loss or harm
  - e.g., a file server that doesn't authenticate its users

# Some terminology

- Threats
  - A loss or harm that might befall a system
  - e.g., users' personal files may be revealed to the public
  - There are four major categories of threats:
    - Interception
    - Interruption
    - Modification
    - Fabrication
  - When designing a system, we need to state the threat model
    - Set of threats we are undertaking to defend against
    - Whom do we want to stop from doing what?

# Some terminology

- **Attack**

- An action which **exploits** a **vulnerability** to **execute** a **threat**
- e.g., telling the file server you are a different user in an attempt to read or modify their files

- **Control**

- Removing or reducing a vulnerability
- You **control** a **vulnerability** to prevent an **attack** and block a **threat**.
- How would you control the file server vulnerability?
- Our goal: control vulnerabilities

# Methods of defence

- How can we defend against a threat?
  - Prevent it: prevent the attack
  - Deter it: make the attack harder or more expensive
  - Deflect it: make yourself less attractive to attacker
  - Detect it: notice that attack is occurring (or has occurred)
  - Recover from it: mitigate the effects of the attack
- Often, we'll want to do many things to defend against the same threat
  - “Defence in depth”

# Example of defence

- Threat: your car may get stolen
- How to defend?
  - Prevent: is it possible to absolutely prevent?
  - Deter: Store your car in a secure parking facility
  - Deflect: Use “The Club”, have sticker mentioning car alarm
  - Detect: Car alarms, OnStar
  - Recover: Insurance

# Defence of computer systems

- Remember we may want to protect any of our **assets**
  - Hardware, software, data
- Many ways to do this; for example:
- Cryptography
  - Protecting data by making it unreadable to an attacker
  - Authenticating users with digital signatures
  - Authenticating transactions with cryptographic protocols
  - Ensuring the integrity of stored data
  - Aid customers' privacy by having their personal information automatically become unreadable after a certain length of time

# Defence of computer systems

- Software controls
  - Passwords and other forms of access control
  - Operating systems separate users' actions from each other
  - Virus scanners watch for some kinds of malware
  - Development controls enforce quality measures on the original source code
  - Personal firewalls that run on your desktop



# Defence of computer systems

- Hardware controls
  - Not usually protection of the hardware itself, but rather using separate hardware to protect the system as a whole.
  - Fingerprint readers
  - Smart tokens
  - Firewalls
  - Intrusion detection systems

# Defence of computer systems

- Physical controls
  - Protection of the hardware itself, as well as physical access to the console, storage media, etc.
  - Locks
  - Guards
  - Off-site backups
  - Don't put your data centre on a fault line in California

# Defence of computer systems

- Policies and procedures
  - Non-technical means can be used to protect against some classes of attack
  - If an employee connects his own Wi-Fi access point to the internal company network, that can accidentally open the network to outside attack.
    - So don't allow the employee to do that!
  - Rules about changing passwords
  - Training in best security practices

# Recap

- What is our goal in this course?
  - Identify security and privacy issues
  - Design systems that are more protective of security and privacy
- What is security?
  - Confidentiality, Integrity, Availability
- What is privacy?
  - Informational self-determination

# Recap

- Who are the adversaries?
  - Learn to think like an attacker
- Assets, vulnerabilities, threats, attacks and controls
  - You **control** a **vulnerability** to prevent an **attack** and block a **threat**.
- Methods of defence
  - Cryptography, software controls, hardware controls, physical controls, policies and procedures