CS4238: Computer Security Practice

Lecture 1-A: Introduction & Administration

Slides by: LIANG Zhenkai,
Roland YAP & SUFATRIO

Module Information (From CORS)

Module Code :	CS4238
Module Title :	Computer Security Practice
Module Description :	This is a practice security module with emphasis on hands-on experiences of computer security. The objective of this module is to connect computer security knowledge to practical skills, including common attacks and protection mechanisms, system administration, and development of secured software. Topics covered include network security, operating system security, and application security, such as DNS attacks, memory-error exploits, ~~*and web application attacks*~~. Students will learn through lab-based exercises and assignments.

Module Information (From CORS)

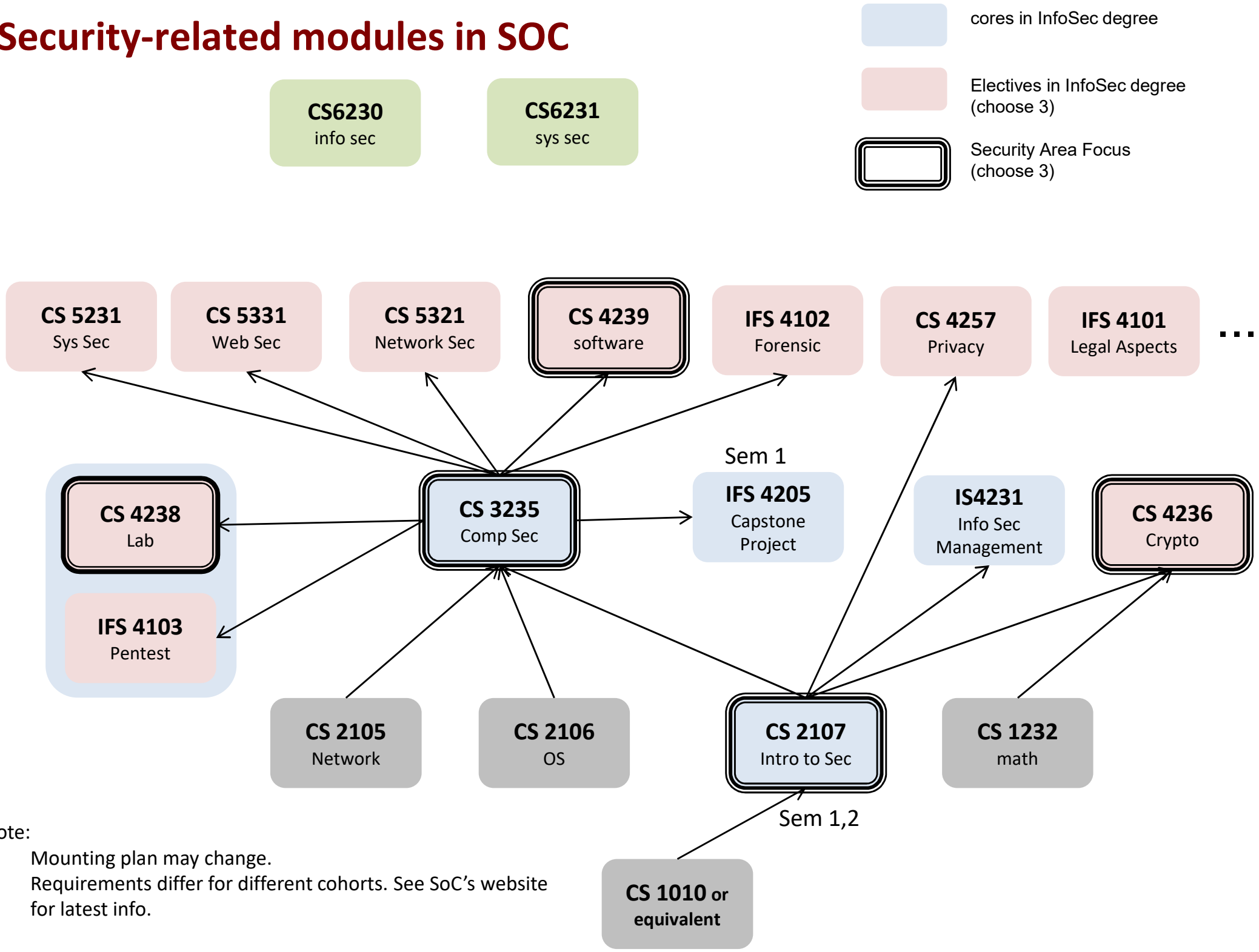
Module Examinable :	-
Exam Date :	No Exam Date.
Modular Credits :	4
Pre-requisite :	CS3235 Computer Security
Preclusion :	Nil
Module Workload (A-B-C-D-E)* :	2-0-1-3-4

* A: no. of lecture hours per week
  B: no. of tutorial hours per week
  C: no. of laboratory hours per week
  D: no. of hours for projects, assignments, fieldwork etc per week
  E: no. of hours for preparatory work by a student per week

About This Module

- **Practice** of system and network security
  - Understanding **security principles through practice**
  - Learning skills of programming, system administration, security tools
    - Only a start – security is ever changing
- **Preparation** for a career in cyber security

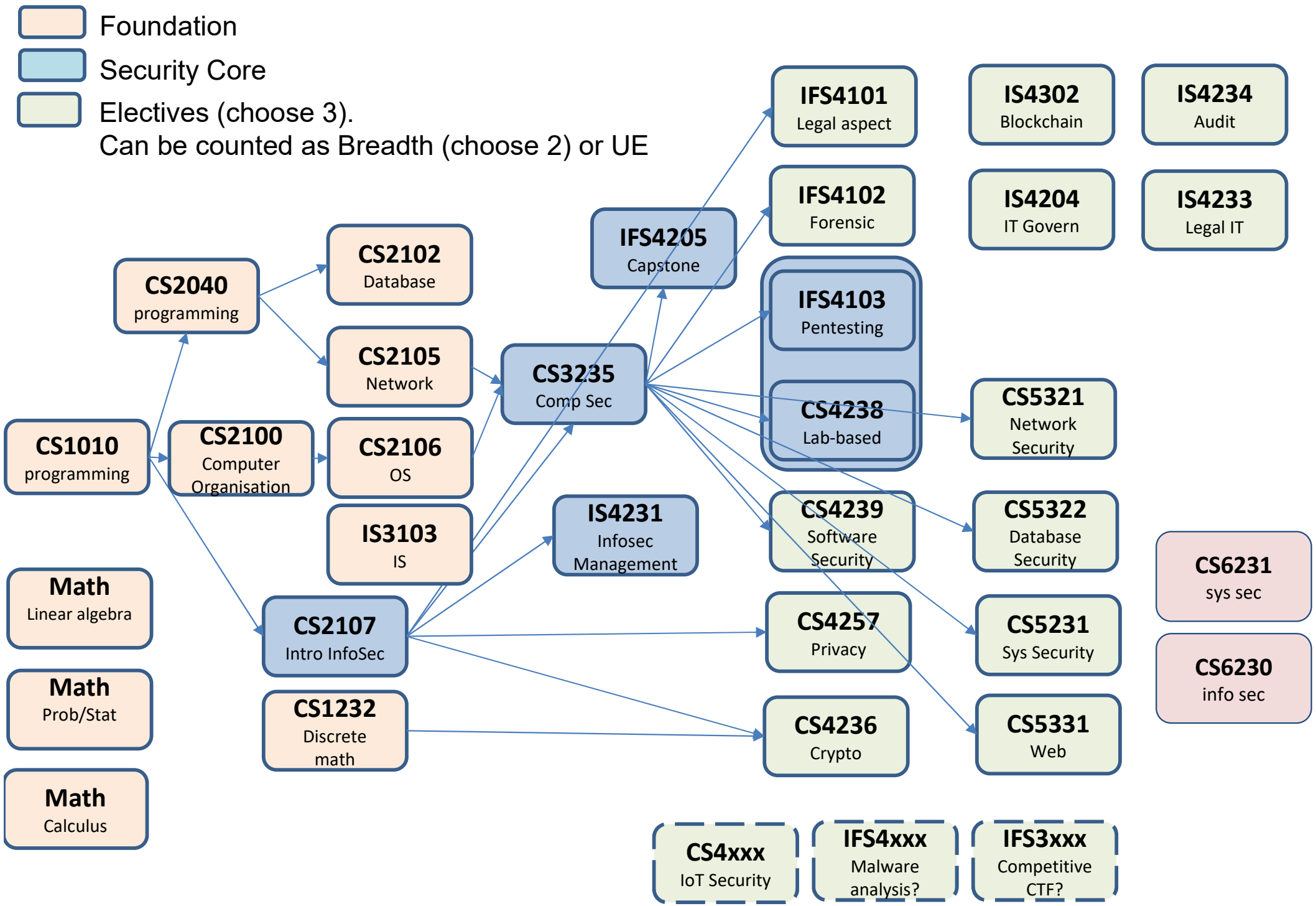
Security-related modules in SOC



Note:

1. Mounting plan may change.
2. Requirements differ for different cohorts. See SoC's website for latest info.

Security-related modules and BCOMP InfoSec requirements



Teaching Staff & Module Resource

Lecturer: Prasanna Karthik Vairam, Sufatrio (Rio)

Co-Lecturer: Adrian Tang (Malware Analysis)

TAs: TBD

CS4238 on Canvas:

- Admin Details and Announcements
- Lecture notes, lab notes, assignments: check it regularly!

MS Teams: additional platform for online interaction and back-up online lectures+labs

Teaching Mode & Grading

- 12 lectures
- 11 labs (starting from Week 2) + *Lab 0*
- No final exam
- Continual Assessment (100%, *tentatively*):
  - Individual assignments (15%+15%+10%+10%): 50%
  - Individual mid-term evaluation (Week **8**): 20%
  - Group-based project report: 30%

Tentative Schedule

CS4238 (Computer Security Practice) AY2022/23: Tentative Schedule

Week No	Date (on Mon)	Lecture Topic	Labs	Asssignments
1	9-Jan	L1: Introduction, Unix/Linux Administration, Networking Overview	L0	
2	16-Jan	L2: Vulnerability Scanning and Automatic Exploitation	L1	
3	23-Jan	*Public Holiday*		
4	30-Jan	L3: Assembly Basics, Buffer overflow attacks and defenses	L2	[A1
5	6-Feb	L4: Password Attacks, Network Configuration, Traffic Analysis	L3	
6	13-Feb	L5: Network Attacks and IDS	L4	A1], [A2
		Recess Week		
7	27-Feb	L6: Web security	L5	
8	6-Mar	Mid-term exam	L6 (VM set-up, self-lab)	A2]
9	13-Mar	Basic static and dynamic binary analysis	L6 + Linux static	[Project, [A3
10	20-Mar	Advanced Static Analysis	L7 + Linux dynamic	
11	27-Mar	Advanced Dynamic Analysis	L8	A3]
12	3-Apr	Malware Behavior Analysis and sample malware analysis	L9	[A4,
13	10-Apr	Project presentations + Review	Project presentations, L10 (self)	
		Reading Week		A4], Project]
	Suggested sessions (TBD)?			

Class Arrangement

- *For now*: synchronous online + recording
  - **Hybrid classes** as below
  - Lectures and Labs at COM1-02-12
  - Lecture: 6.30-8.30 PM
  - Lab: 8.30-9.30 PM
  - Zoom: https://nus-sg.zoom.us/j/83727556494?pwd=WDF6TkITOWZ0VWFtWWZISE00clg4dz09
  - Recordings will be made available on a best-effort basis!
    - Remind the instructor to record. ☺

Group Formation

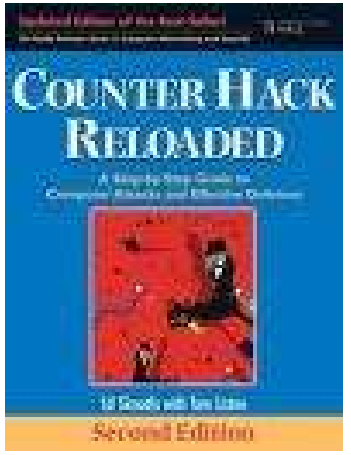
- For your **lab** and **final group-based project**:
  - A group of 2-3 students
  - Your group will remain fixed throughout the module
- Group formation:
  - To be fixed in Week 3
  - Please self-form your group
- Learn and work together in group:
  - Forming → storming → norming → ***performing*** → adjourning

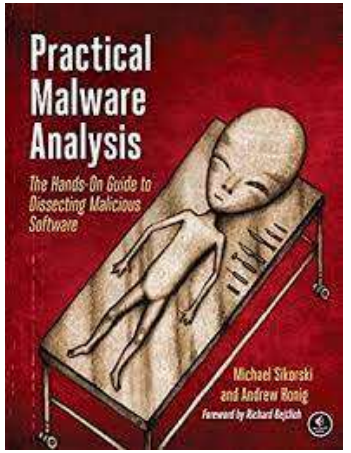
Our Lab Environment

- Most activities use **VMs** on your laptop

Textbooks

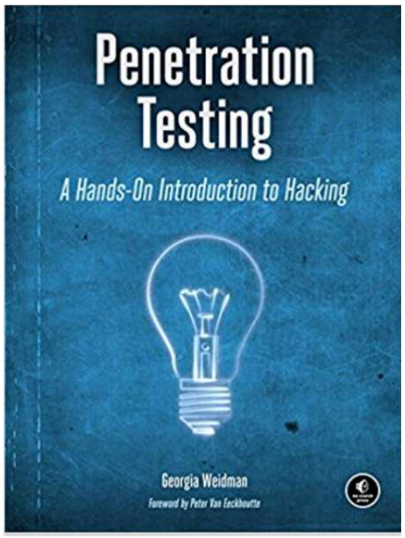
- Reference books:
  - "*Counter Hack Reloaded: A Step-by-Step Guide to Computer Attacks and Effective Defenses*", by Ed Skoudis & Tom Liston (2006)
  - "*Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software*", by Michael Sikorski (2012)
  - Both books are available in our library RBR





Textbooks

- Supplementary book:
  - "*Penetration Testing: A Hands-On Introduction to Hacking*", by Georgia Weidman



- Other supplementary information:
  - See papers + web links
  - Practice: security is always changing

Notes on Assignments

- **Avoid** plagiarism: group study is fine, but do not copy answers
- ChatGPT, Github Copilot and others.
- Canvas and MS Teams channel:
  - For discussions on assignments:
    - You can ask questions and share ideas
    - *But* don't reveal your answers!
  - Please be courteous, even when disagreeing with others

Plagiarism Policy and Guidance Note Changes

"I. The University is taking a tougher stance against academic dishonesty. As such, for cases of plagiarism and cheating in **tests/examinations/graded assignments that have been assessed to be 'Moderate' in severity, the minimum penalty would be a 'Fail' grade for the affected module**.

II. The online version of the revised NUS Plagiarism Policy and Guidance Note can be accessed via the Student Portal.

2 NUS students are expected to uphold the highest standards of academic integrity and honesty at all times, as well as embrace diversity and show mutual respect for one another, both within the University and the wider Singapore community. Students who do not comply with the NUS Statutes and Regulations will face disciplinary action.

3 If you have any queries, suggestions or feedback, please email us at studentconduct@nus.edu.sg."

NUS' Latest Plagiarism Policy

- You should be aware of the consequence: F
- Importance of **academic honesty**
- The module recognizes that **some interactions** with classmates/others can facilitate understanding of the course's material
- The key is: "**be reasonable**"
- We will adopt a policy similar to Harvard CS50's: https://cs50.harvard.edu/x/2020/honesty/
- Next are rules of thumb that (inexhaustively) lists acts that the module considers **reasonable** (based on https://cs50.harvard.edu/x/2020/honesty/)

Still Reasonable in Our Module

- **Discussing the course's material** or assignment task with others in order to understand it better
- Whiteboarding solutions with others **using diagrams or pseudocode** but not actual code
- **Sending or showing code that you've written to someone**, possibly a classmate, so that they might help you identify and fix a bug
- **Incorporating a few lines of code that you find online** or elsewhere into your own code, provided that those lines are not themselves solutions to assigned work and that **you cite the lines' origins**
- Turning to the web or elsewhere for instruction beyond the course's own, for references, and for solutions to technical difficulties, but not for outright solutions to assigned work

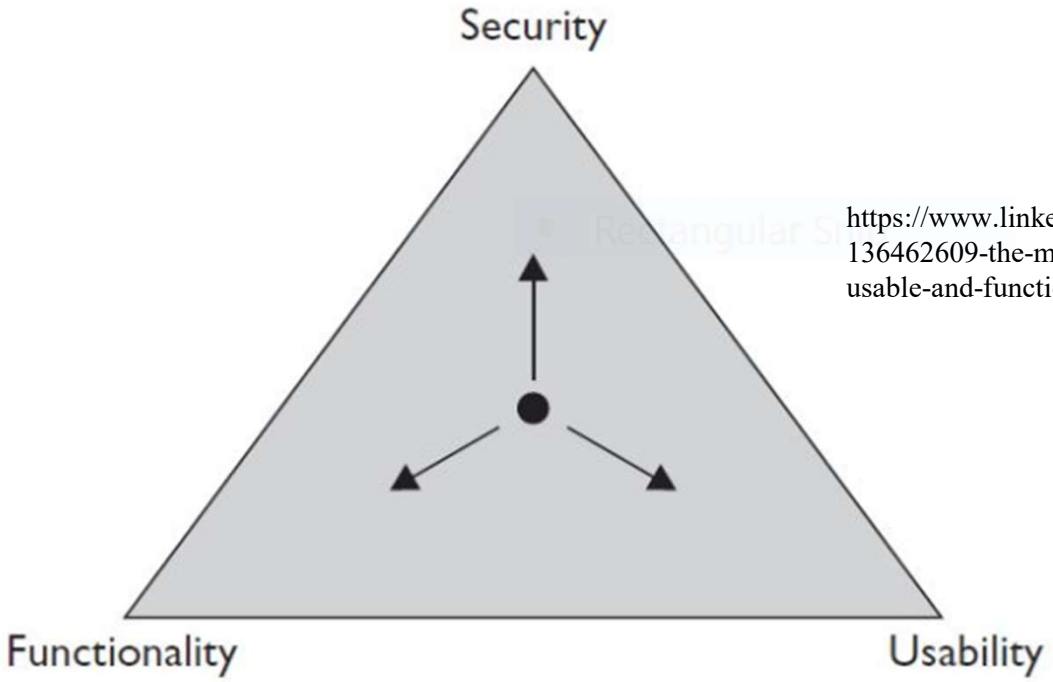
Computer Security Problems

Why is There a Big Security Problem?

- Functionality: the primary concern during design and implementation
  - Security is a secondary goal
    - Features pay the bills (typically)
  - Not aware/familiar with security problems
- Unavoidable human mistakes
  - (Lack of) awareness
  - Lazy programmer
- Complex modern computing systems
  - Large attack surface, e.g. Windows

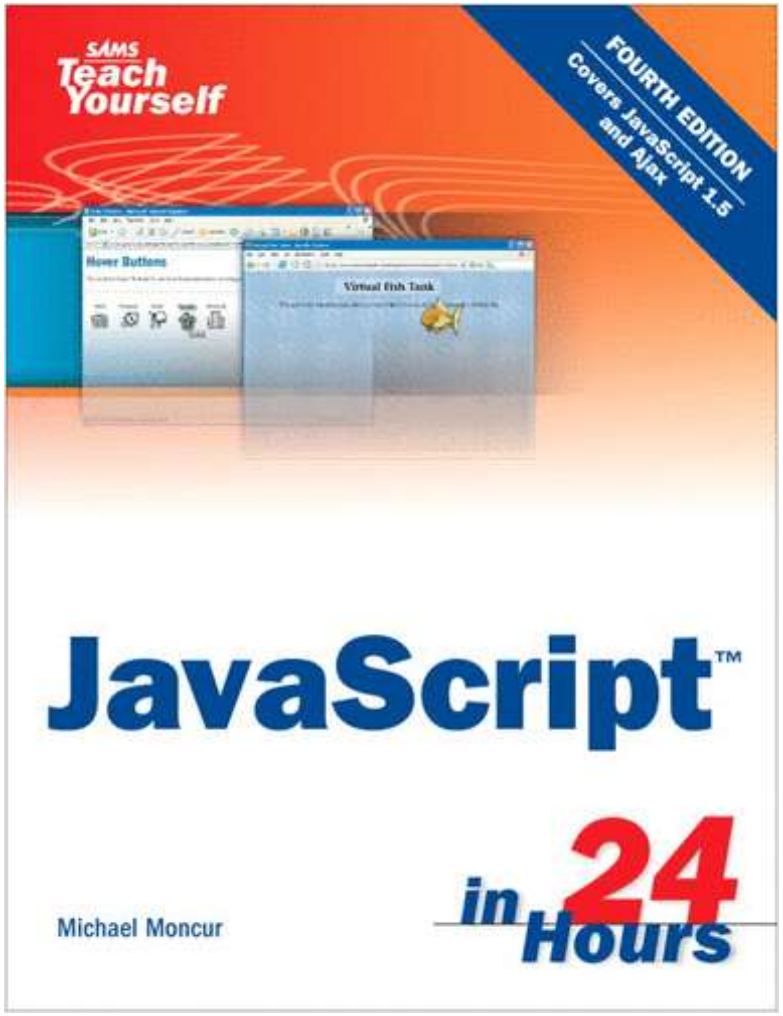
Why is There a Big Security Problem?

- *Security*, *Functionality* and *Ease-of-Use Triangle*: the more secure something is, the less usable and functional it becomes



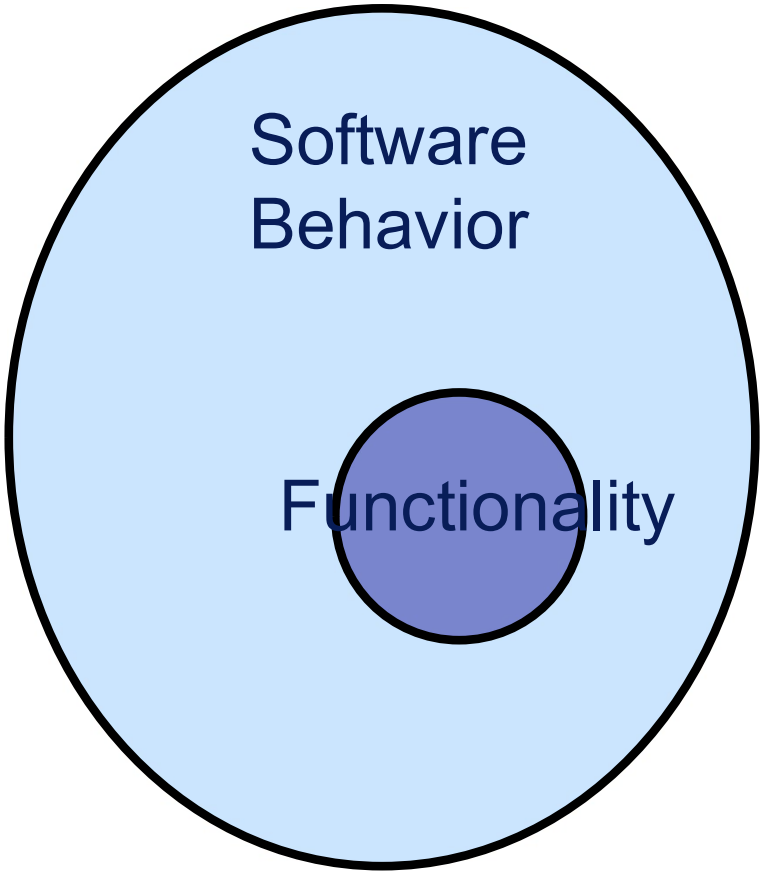
https://www.linkedin.com/pulse/20140619200426-136462609-the-more-secure-something-is-the-less-usable-and-functional-it-becomes

Impatient Programmers



- Maybe enough for learning basic functionality
- Never enough for learning subtle implications of functionalities
- Result: programs can do more than you expect

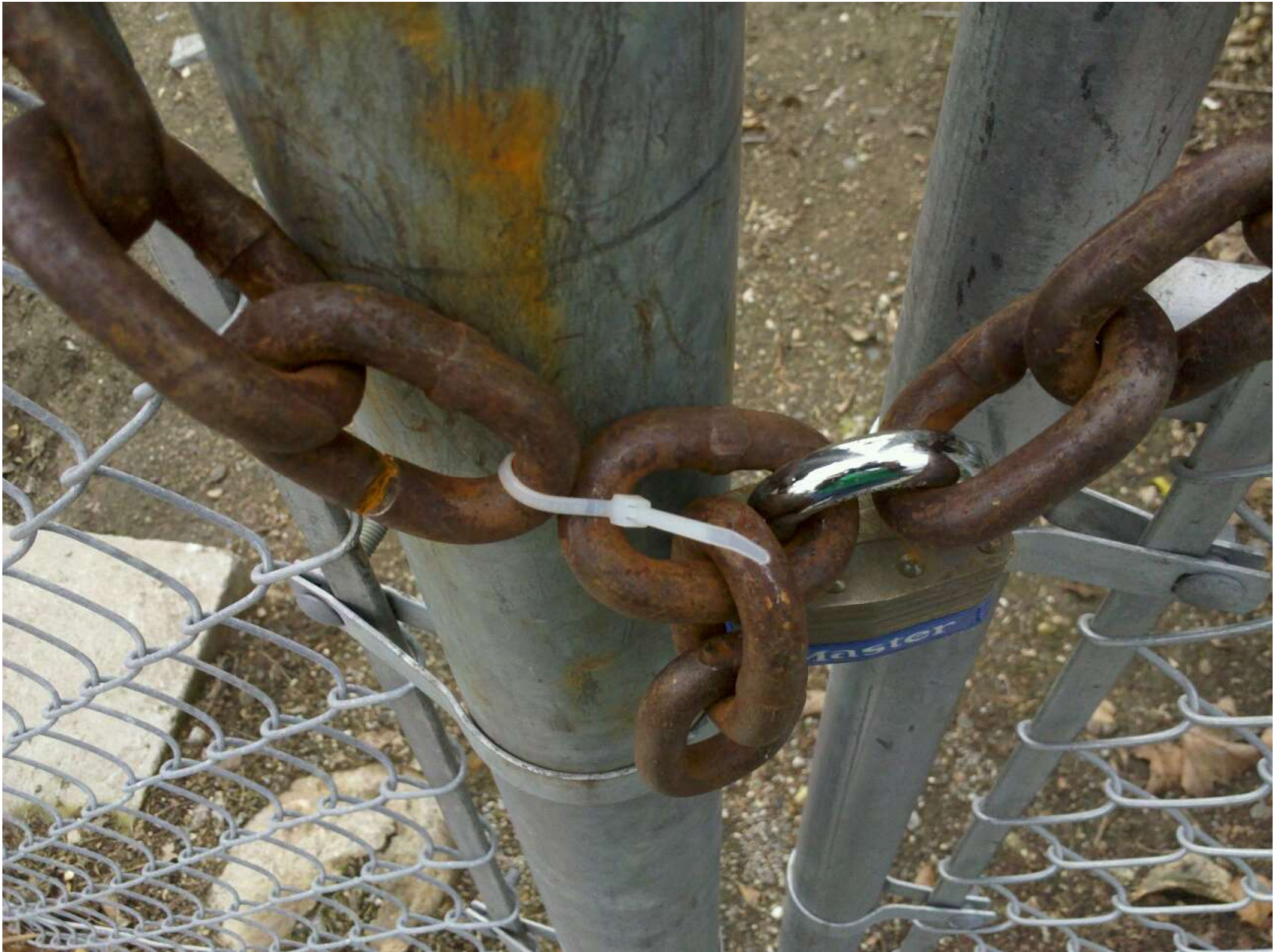
Security: Mission Impossible



- Cyber security
  - Warfare between attackers & defenders
  - Advantage on attackers (why?)
- But in practice, we still manage the security problem
  - At least partially
- Need a better understanding of the whole system
- Why?

Principle of Easiest Penetration

- Security is about **every** aspect of a computing system
  - Hardware, software, data and people
- Principle of easiest penetration:
  - Any system is **most** vulnerable at its **weakest point**
  - Attackers **don't** follow any rules. Never underestimate their creativity



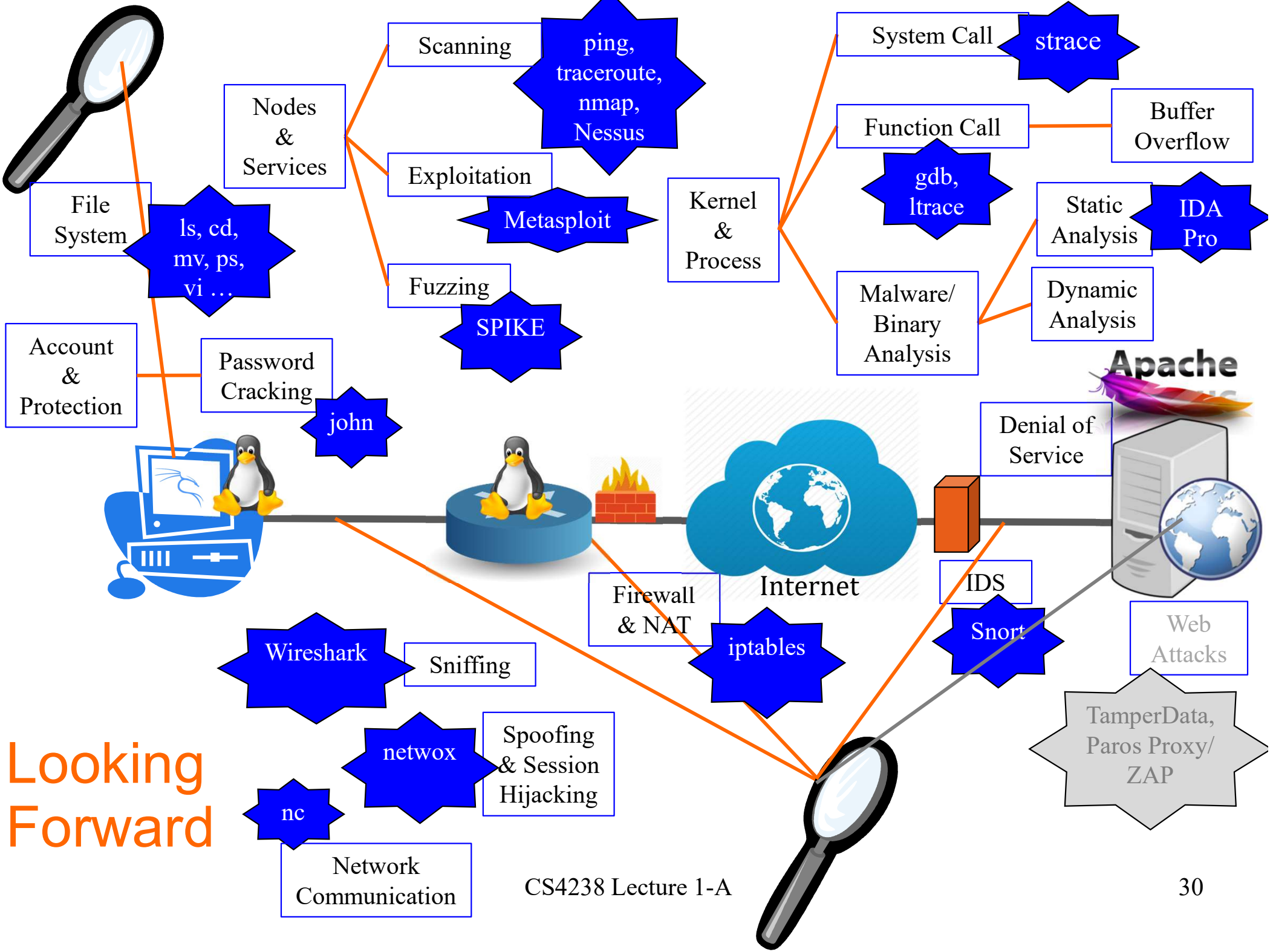
Think Like an Attacker

- Adversary is targeting your **assets**, not your defenses
- Will try to exploit the **weakest** part of your defenses
  - E.g. bribe human operator,
    social engineering,
    (physically) steal server containing data

In This Module

Sample Topics

- Linux (Kali & Ubuntu) & VM
- Attack framework, reconnaissance
- Scanning and network fuzzing
- Memory exploits and defense
- Network attacks and defense
- Malware/binary analysis:
  - Basic static & dynamic analyses
  - Advanced static & dynamic analyses
  - (Windows) malware behavior analysis


Scanning
ping, traceroute, nmap, Nessus
Nodes & Services
Exploitation
Metasploit
Fuzzing
SPIKE
File System
ls, cd, mv, ps, vi …
Account & Protection
Password Cracking
john
Kernel & Process
System Call
strace
Function Call
Buffer Overflow
gdb, ltrace
Static Analysis
IDA Pro
Malware/ Binary Analysis
Dynamic Analysis
Apache
Denial of Service
Internet
Firewall & NAT
iptables
IDS
Snort
Web Attacks
TamperData, Paros Proxy/ ZAP
Wireshark
Sniffing
netwox
Spoofing & Session Hijacking
nc
Network Communication
Looking Forward

Ethical Issue

Learning to Attack

- If you know the enemy and know yourself, you need not fear the result of a hundred battles.

知己知彼，百战不殆。

Sun Tzu, Art of War

- To prevent attack, we need to learn how attack happens

Actors

- Users
- Black hats:
  - Exploit vulnerability
- White hats:
  - Only use vulnerability with permission
- Grey hats:
  - In between white and black hats

Ethical Use of Security Information

- We discuss vulnerabilities and attacks
  - Most vulnerabilities have been fixed
    - But do not assume that systems **have been** patched/fixed
    - Some attacks may still cause harm!
  - Do *not* try these at home
    - Do not try on NUS network
    - Work in a controlled environment
    - Don not launch attacks on external sites from the lab!
    - The case of Morris Worm.
- Purpose of this class
  - Learn to prevent malicious **attacks**
  - Use knowledge for **good** purposes

Vulnerability Disclosure

- *If you find a vulnerability, what should you do?*
  - No disclosure:
    bad as 0-day attacks can exploit the vulnerability
  - Arbitrary disclosure: opens vulnerability time window
- How to disclose the vulnerability?
- **Responsible** vulnerability disclosure:
  - Disclose vulnerability to vendor/stakeholders,
    and allow a time period to find mitigation or patch
  - The problem is: how long?
    - Google Project Zero: 90-day window given before
      a vulnerability is publicly disclosed

Singapore Law

- Computer Misuse and Cybersecurity Act
  - Read the Act – latest 2013
  - Defines unauthorised access
  - Offences: fines/jail
  - Intent to commit
    - Immaterial if authorised/unauthorised
    - Covers: modification/use/interception/obstruction
  - Scope
    - Any person inside/outside Singapore (nationality/citizenship)
  - National security provisions
  - Arrest without warrant

Don’t Cross the Yellow/Red Line

Summary

- Learning principles through practice
  - Seeing is believing

- Practical skills
  - Experience with Linux and open source tools
  - Solutions for your new concerns

Questions?