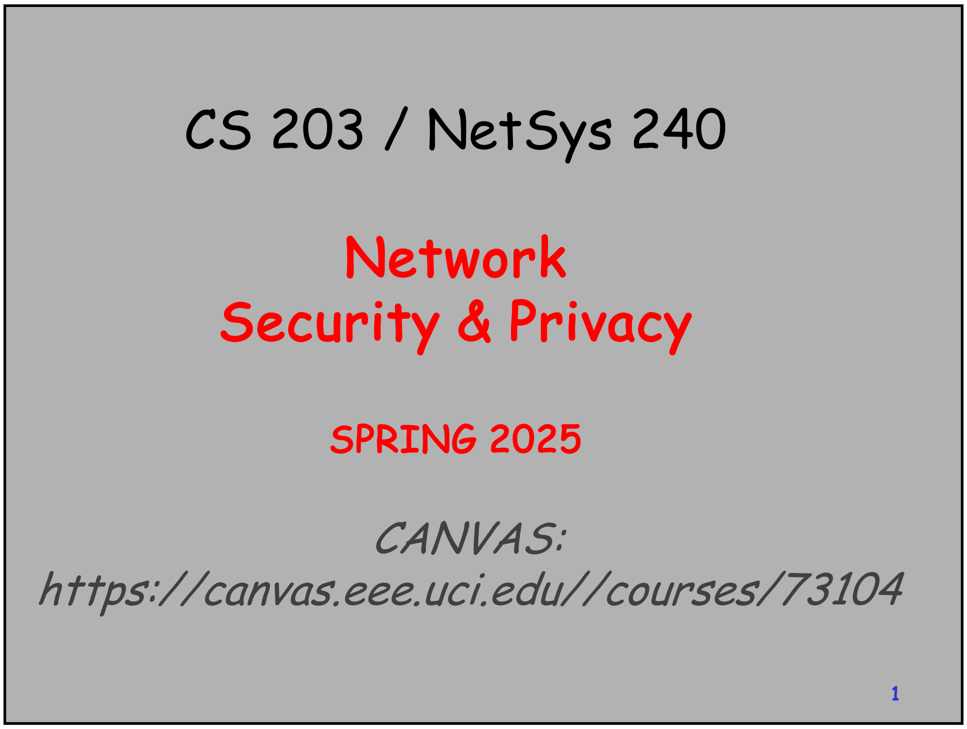
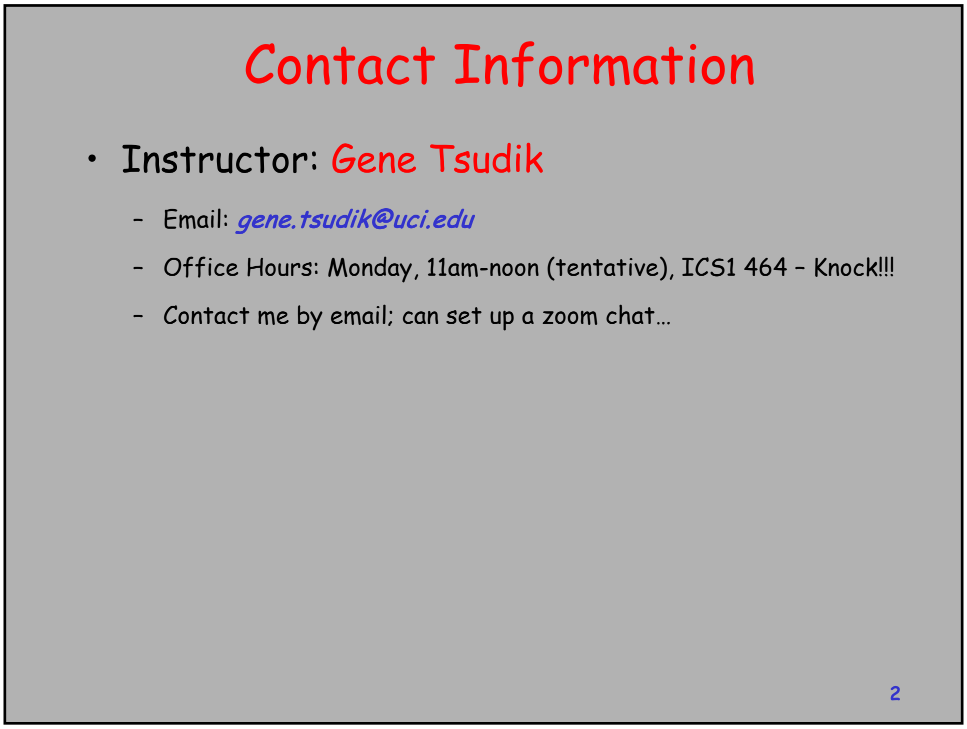


CS 203 / NetSys 240

Network Security & Privacy

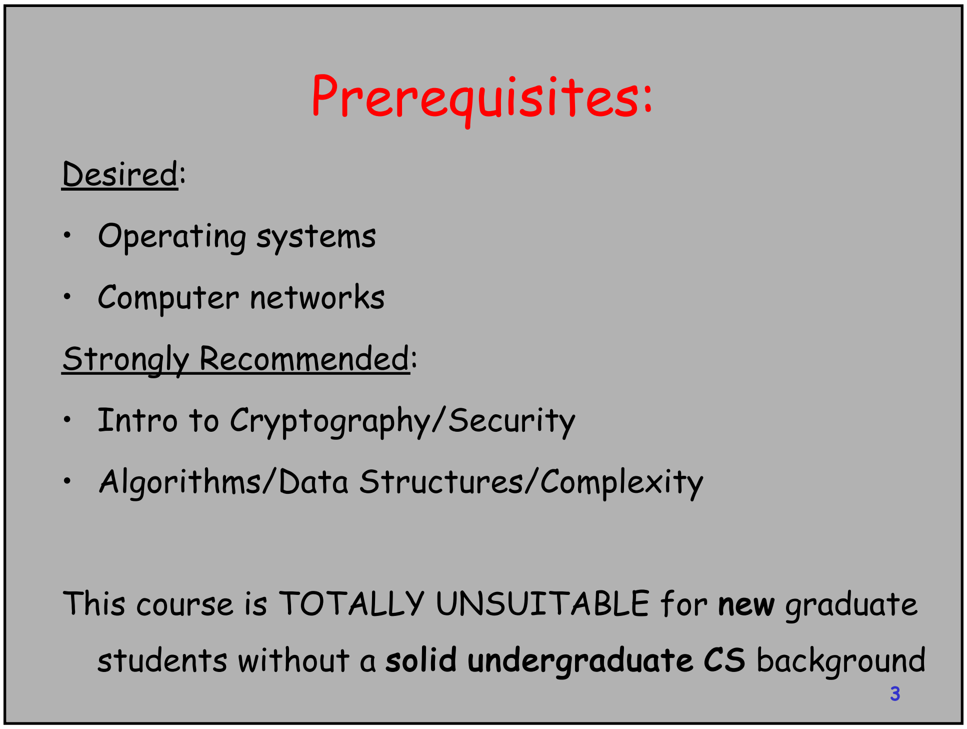
SPRING 2025

CANVAS: https://canvas.eee.uci.edu//courses/73104



Contact Information

- Instructor: Gene Tsudik
  - Email: ***gene.tsudik@uci.edu***
  - Office Hours: Monday, 11am-noon (tentative), ICS1 464 – Knock!!!
  - Contact me by email; can set up a zoom chat…



Prerequisites:

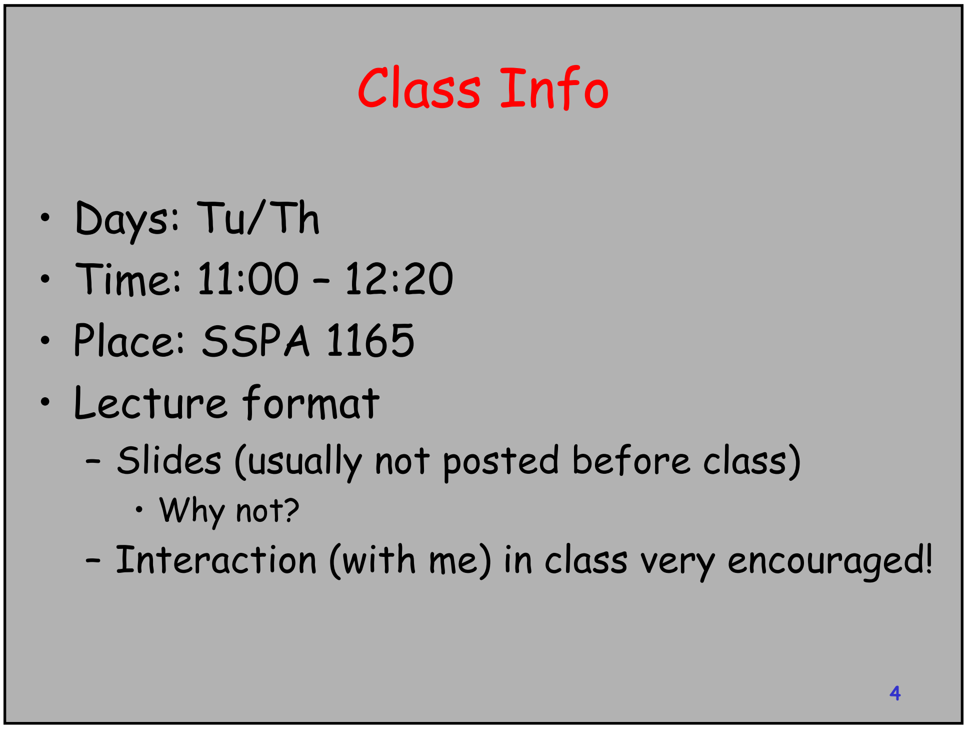
Desired:

- Operating systems
- Computer networks

Strongly Recommended:

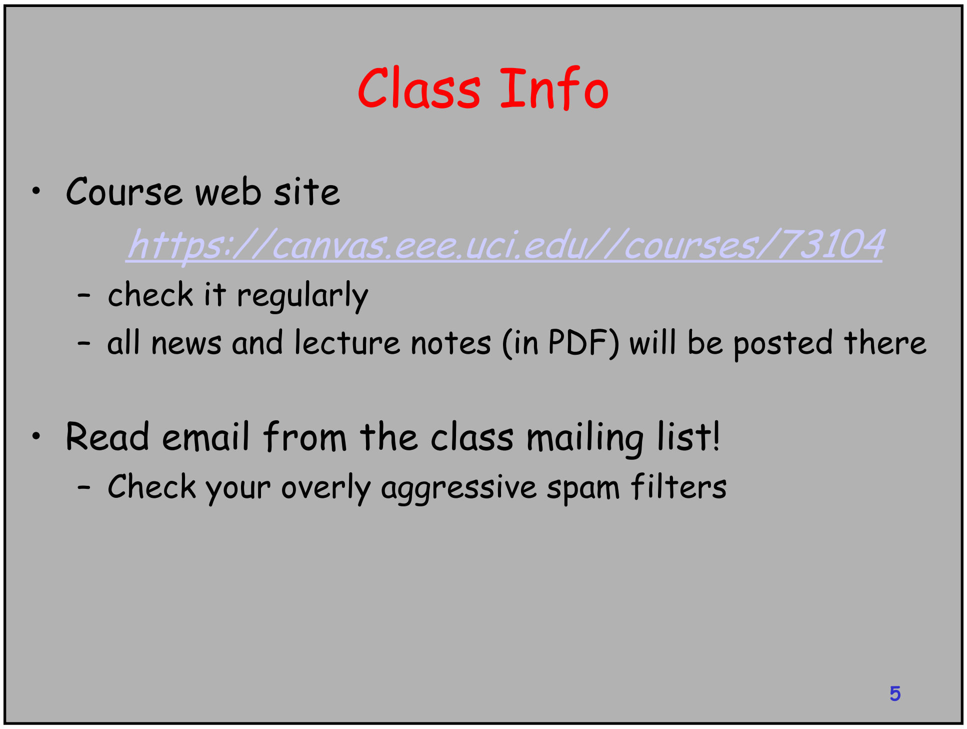
- Intro to Cryptography/Security
- Algorithms/Data Structures/Complexity

This course is TOTALLY UNSUITABLE for **new** graduate students without a **solid undergraduate CS** background



Class Info

- Days: Tu/Th
- Time: 11:00 – 12:20
- Place: SSPA 1165
- Lecture format
  - Slides (usually not posted before class)
    - Why not?
  - Interaction (with me) in class very encouraged!



Class Info

- Course web site

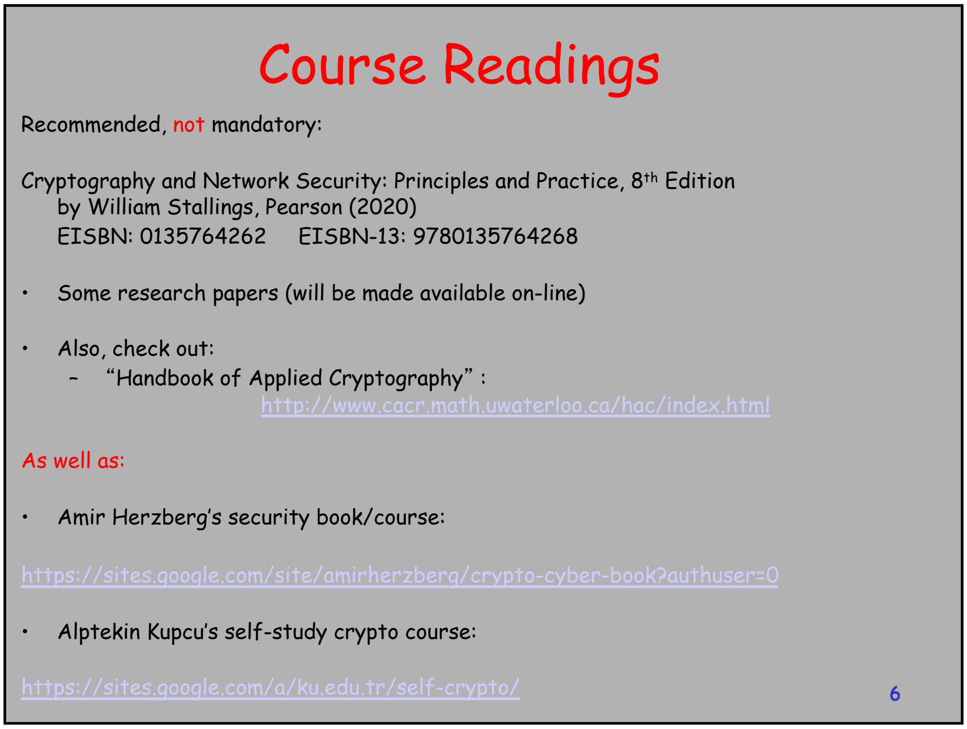
  https://canvas.eee.uci.edu//courses/73104

  – check it regularly

  – all news and lecture notes (in PDF) will be posted there

- Read email from the class mailing list!

  – Check your overly aggressive spam filters



Course Readings

Recommended, not mandatory:

Cryptography and Network Security: Principles and Practice, 8th Edition
by William Stallings, Pearson (2020)
EISBN: 0135764262 EISBN-13: 9780135764268

- Some research papers (will be made available on-line)

- Also, check out:

  – “Handbook of Applied Cryptography” :

  http://www.cacr.math.uwaterloo.ca/hac/index.html

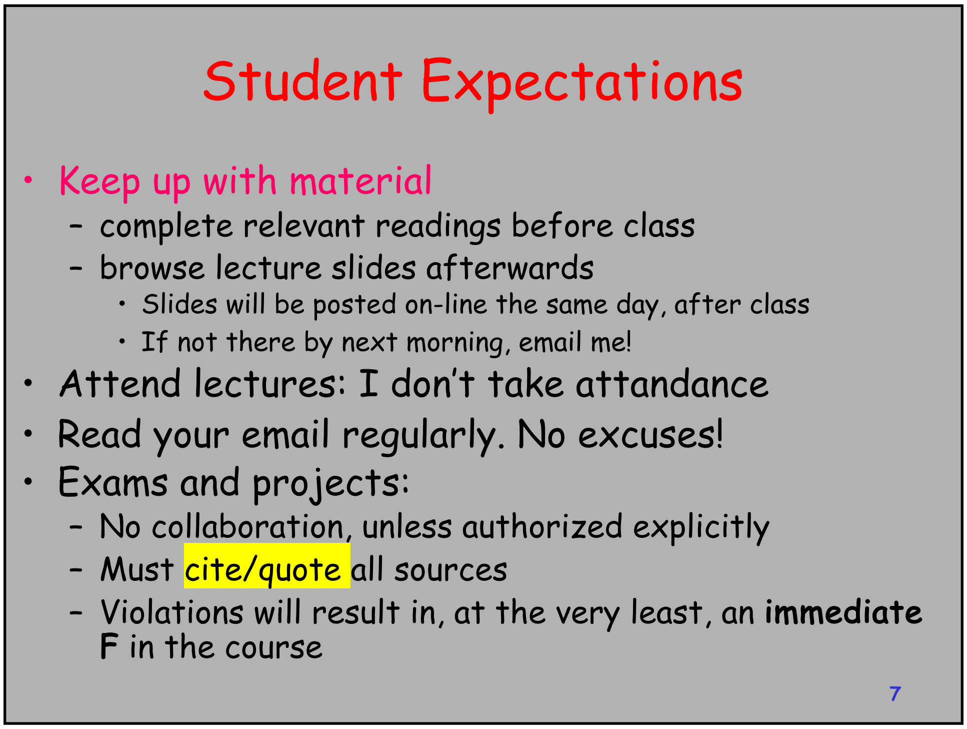
As well as:

- Amir Herzberg’s security book/course:

https://sites.google.com/site/amirherzberg/crypto-cyber-book?authuser=0

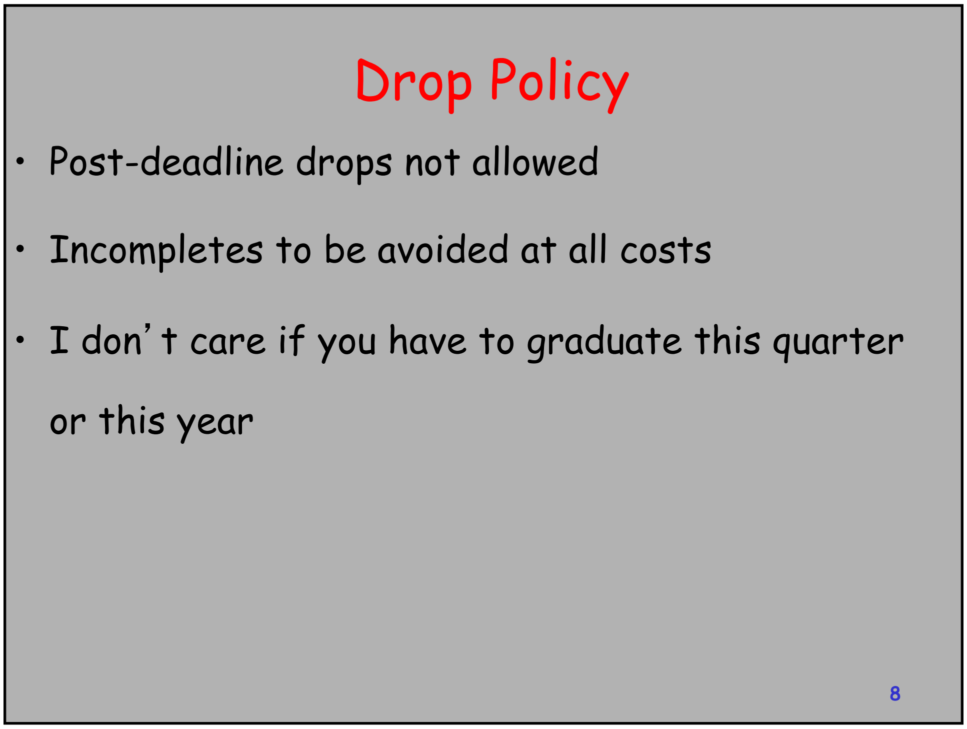
- Alptekin Kupcu’s self-study crypto course:

https://sites.google.com/a/ku.edu.tr/self-crypto/



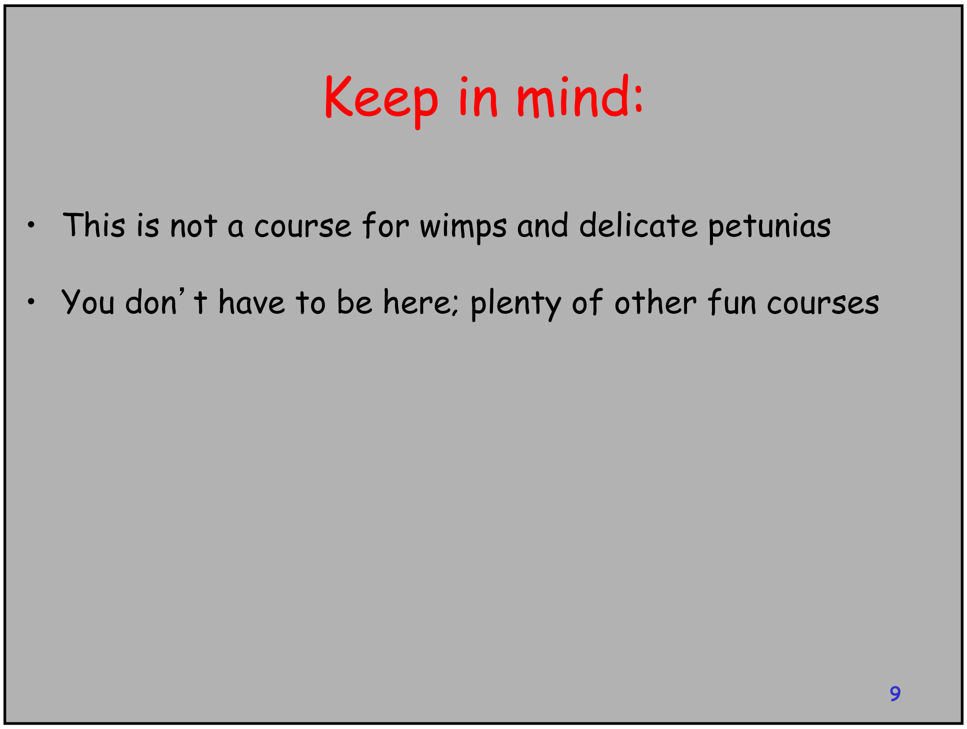
Student Expectations

- Keep up with material
  - complete relevant readings before class
  - browse lecture slides afterwards
    - Slides will be posted on-line the same day, after class
    - If not there by next morning, email me!
- Attend lectures: I don’t take attandance
- Read your email regularly. No excuses!
- Exams and projects:
  - No collaboration, unless authorized explicitly
  - Must cite/quote all sources
  - Violations will result in, at the very least, an **immediate F** in the course



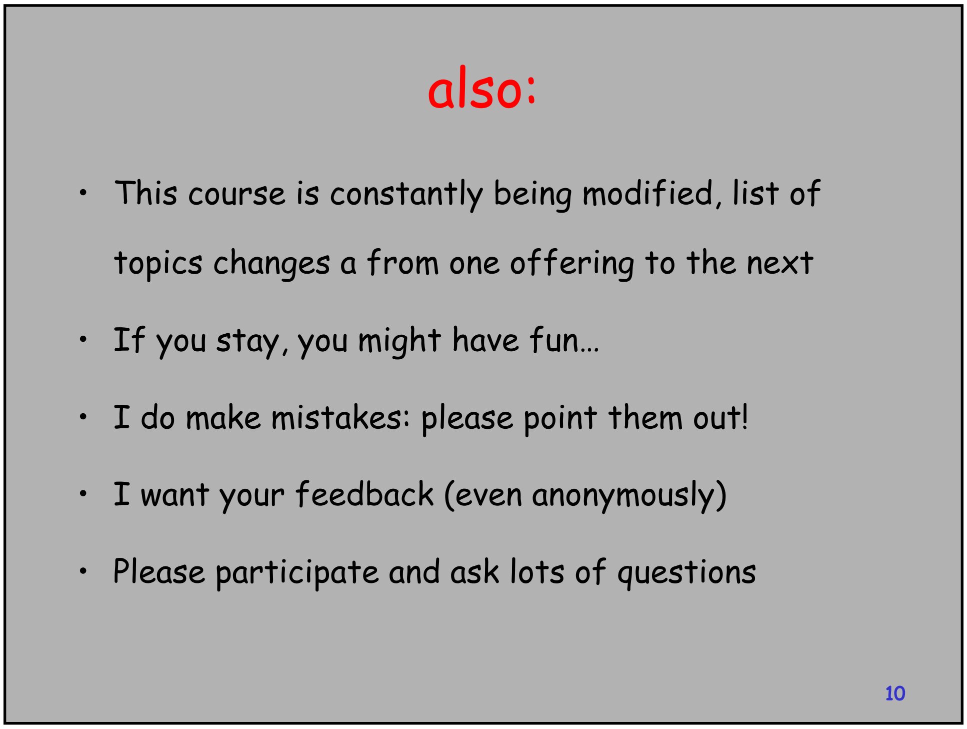
Drop Policy

- Post-deadline drops not allowed
- Incompletes to be avoided at all costs
- I don’t care if you have to graduate this quarter or this year



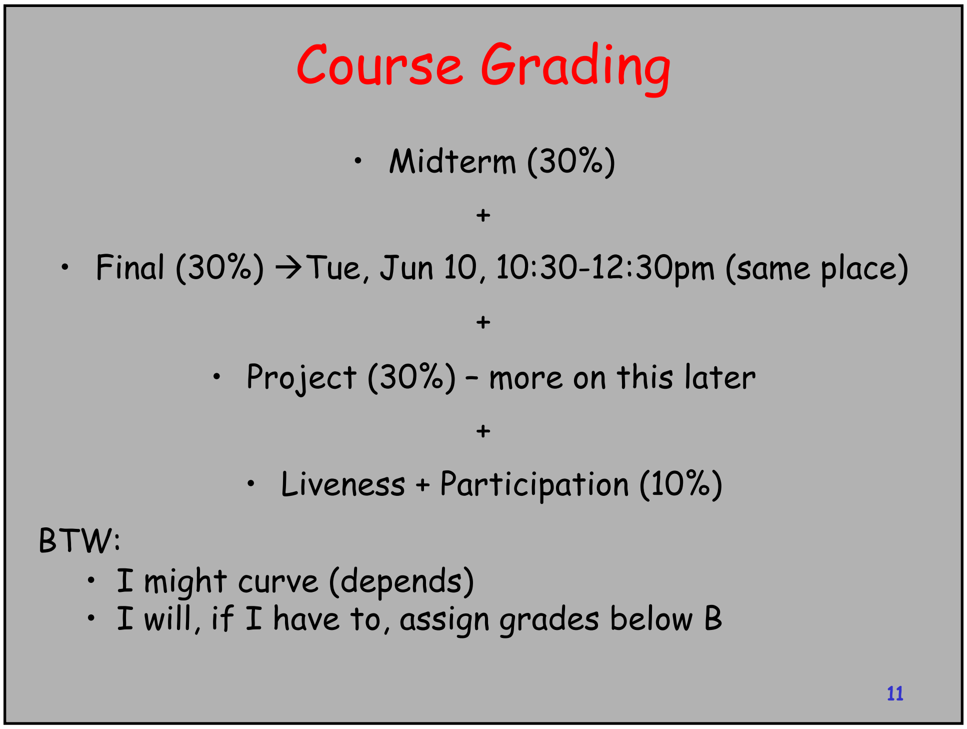
Keep in mind:

- This is not a course for wimps and delicate petunias
- You don't have to be here; plenty of other fun courses



also:

- This course is constantly being modified, list of topics changes a from one offering to the next
- If you stay, you might have fun…
- I do make mistakes: please point them out!
- I want your feedback (even anonymously)
- Please participate and ask lots of questions



Course Grading

- Midterm (30%)

+

- Final (30%) →Tue, Jun 10, 10:30-12:30pm (same place)

+

- Project (30%) – more on this later

+

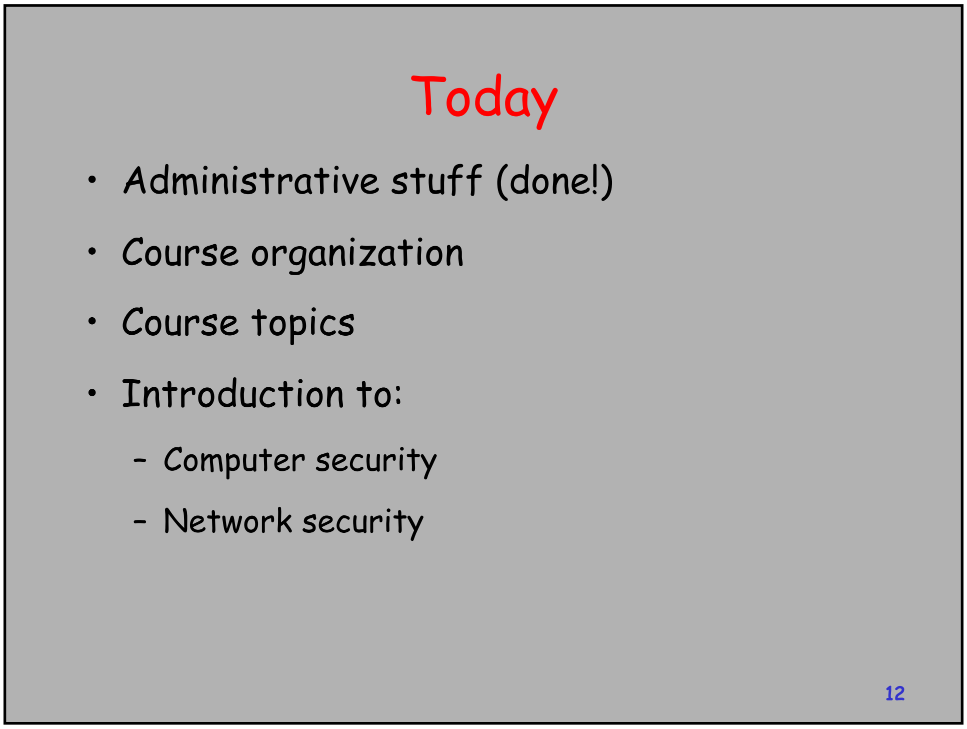
- Liveness + Participation (10%)

BTW:

- I might curve (depends)
- I will, if I have to, assign grades below B

11

11

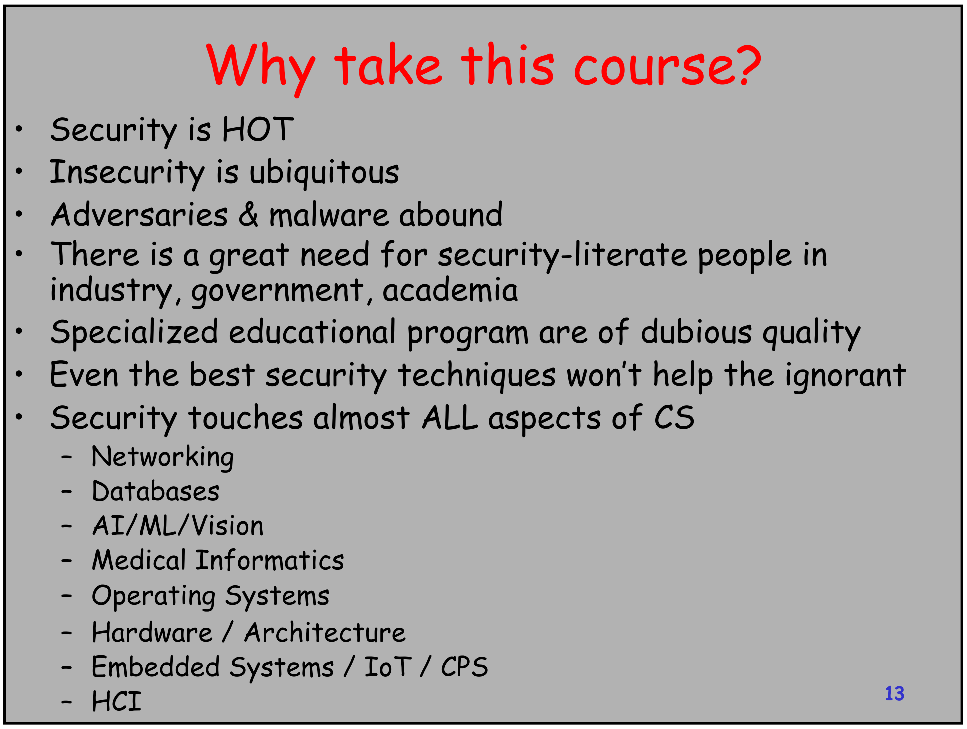


Today

- Administrative stuff (done!)
- Course organization
- Course topics
- Introduction to:
  - Computer security
  - Network security

12

12

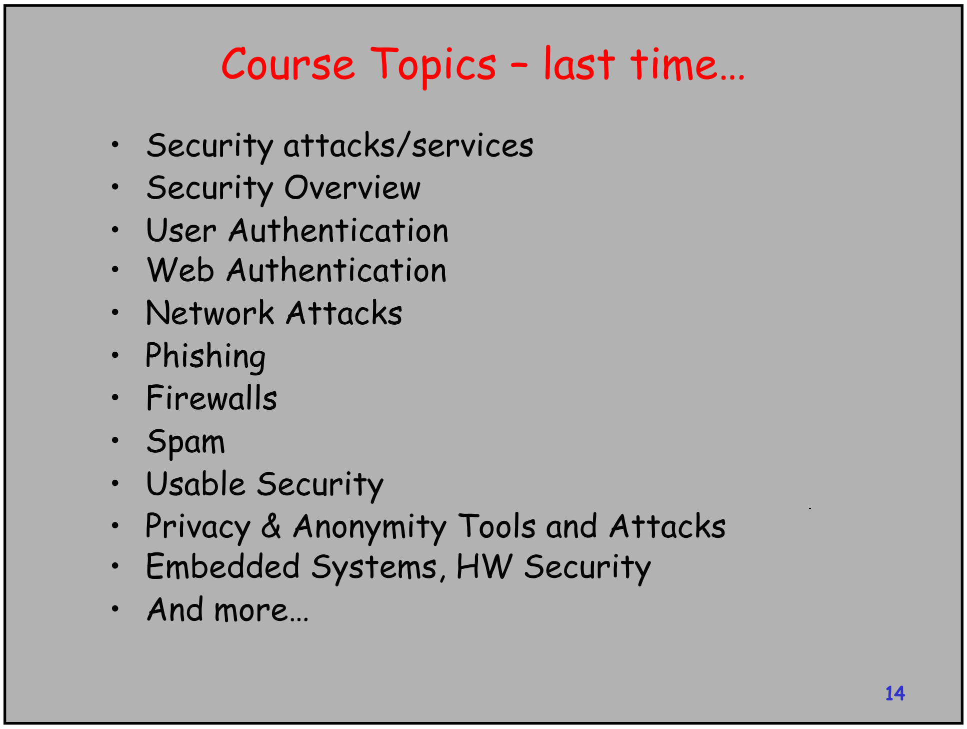


Why take this course?

- Security is HOT
- Insecurity is ubiquitous
- Adversaries & malware abound
- There is a great need for security-literate people in industry, government, academia
- Specialized educational program are of dubious quality
- Even the best security techniques won't help the ignorant
- Security touches almost ALL aspects of CS
  - Networking
  - Databases
  - AI/ML/Vision
  - Medical Informatics
  - Operating Systems
  - Hardware / Architecture
  - Embedded Systems / IoT / CPS
  - HCI

13

13

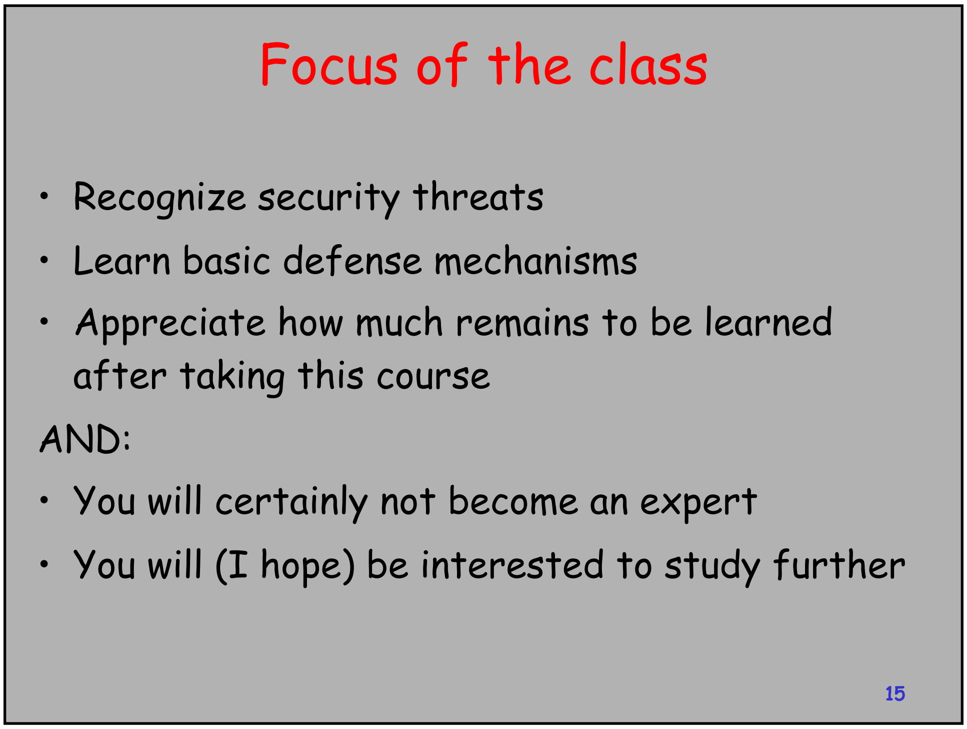


Course Topics – last time…

- Security attacks/services
- Security Overview
- User Authentication
- Web Authentication
- Network Attacks
- Phishing
- Firewalls
- Spam
- Usable Security
- Privacy & Anonymity Tools and Attacks
- Embedded Systems, HW Security
- And more…

14

14



Focus of the class

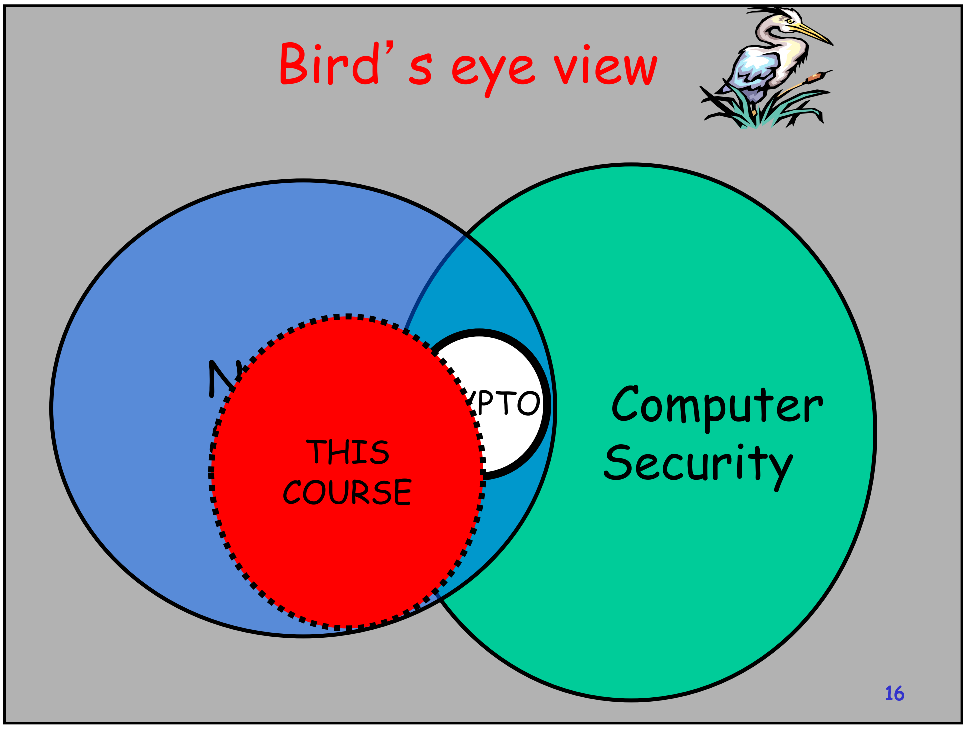
- Recognize security threats
- Learn basic defense mechanisms
- Appreciate how much remains to be learned after taking this course

AND:

- You will certainly not become an expert
- You will (I hope) be interested to study further

15

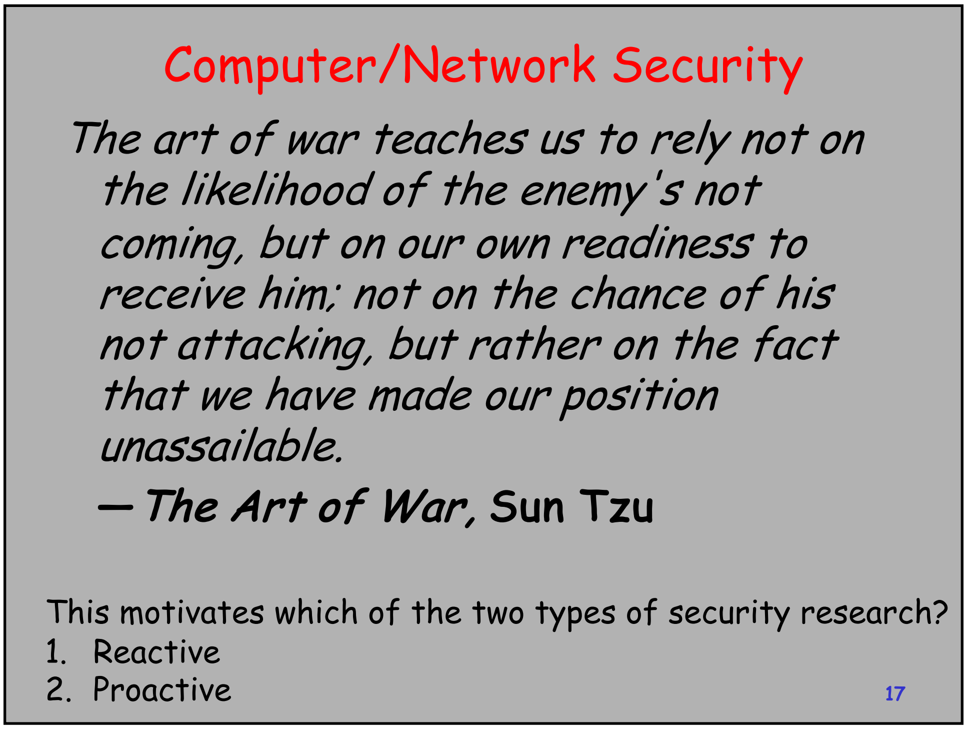
15



Bird's eye view

16

16

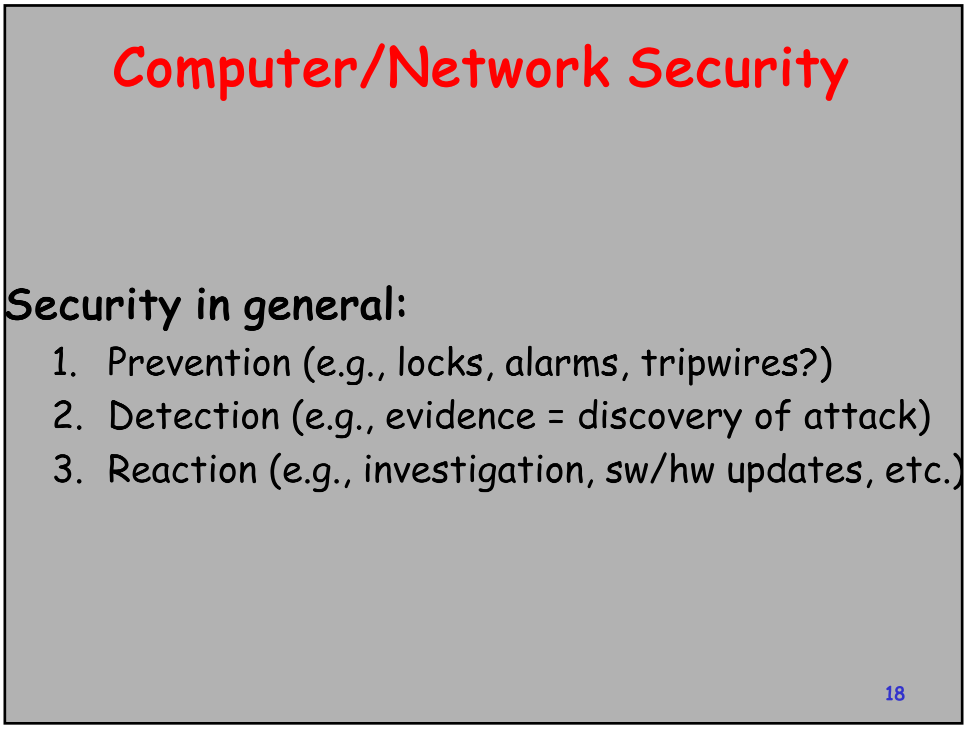


Computer/Network Security

> *The art of war teaches us to rely not on the likelihood of the enemy's not coming, but on our own readiness to receive him; not on the chance of his not attacking, but rather on the fact that we have made our position unassailable.*
>
> **— *The Art of War*, Sun Tzu**

This motivates which of the two types of security research?

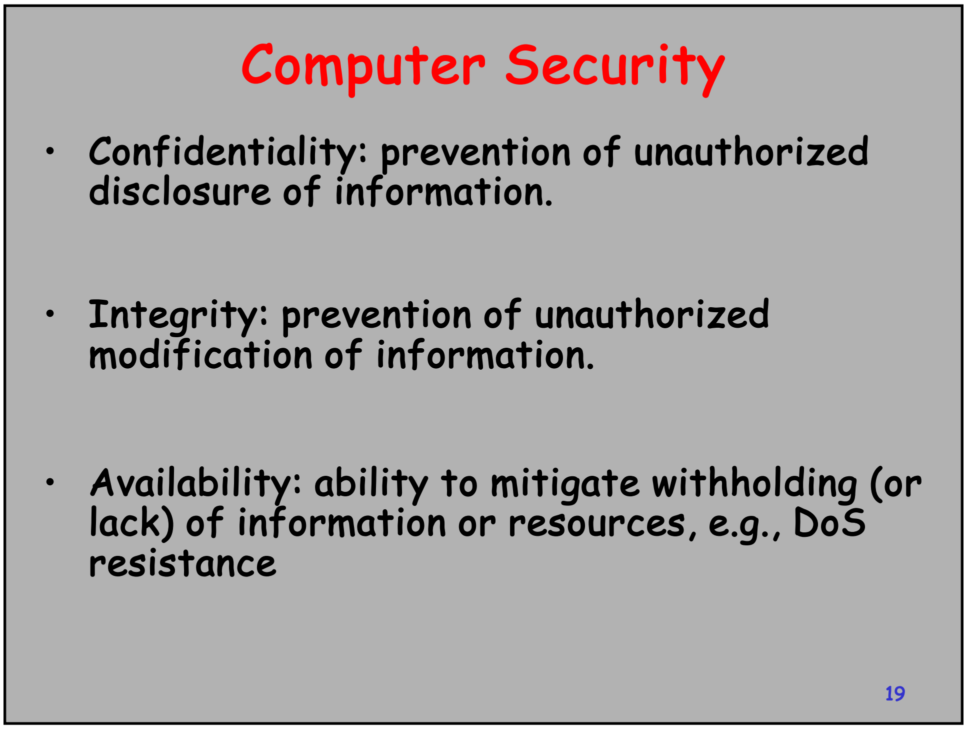
1. Reactive
2. Proactive



Computer/Network Security

Security in general:

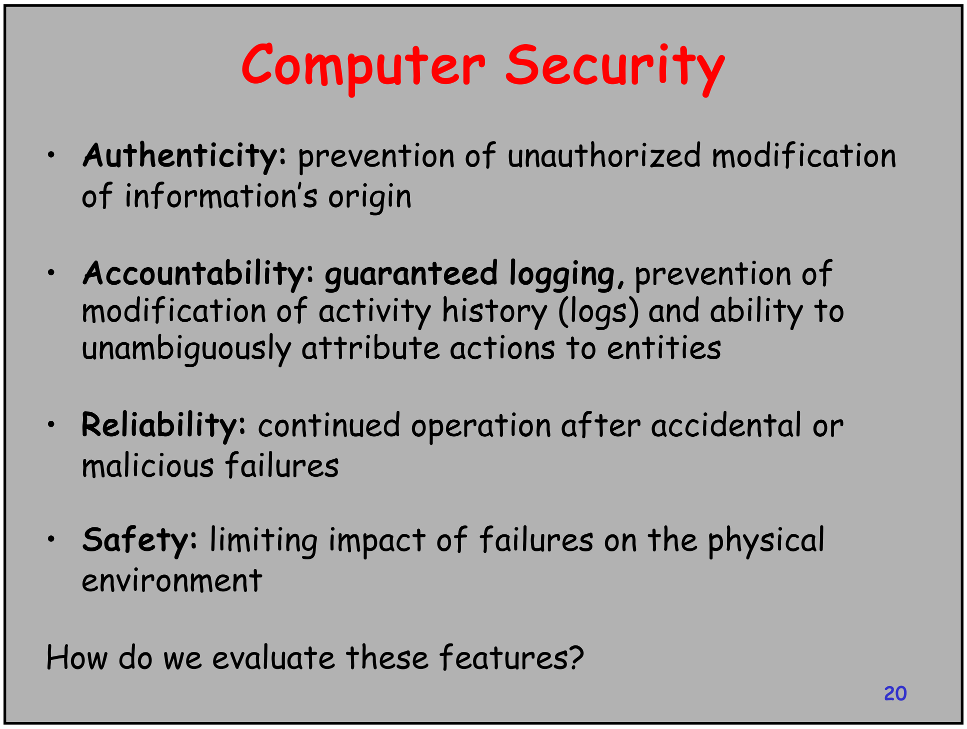
1. Prevention (e.g., locks, alarms, tripwires?)
2. Detection (e.g., evidence = discovery of attack)
3. Reaction (e.g., investigation, sw/hw updates, etc.)



Computer Security

- **Confidentiality: prevention of unauthorized disclosure of information.**
- **Integrity: prevention of unauthorized modification of information.**
- **Availability: ability to mitigate withholding (or lack) of information or resources, e.g., DoS resistance**

19

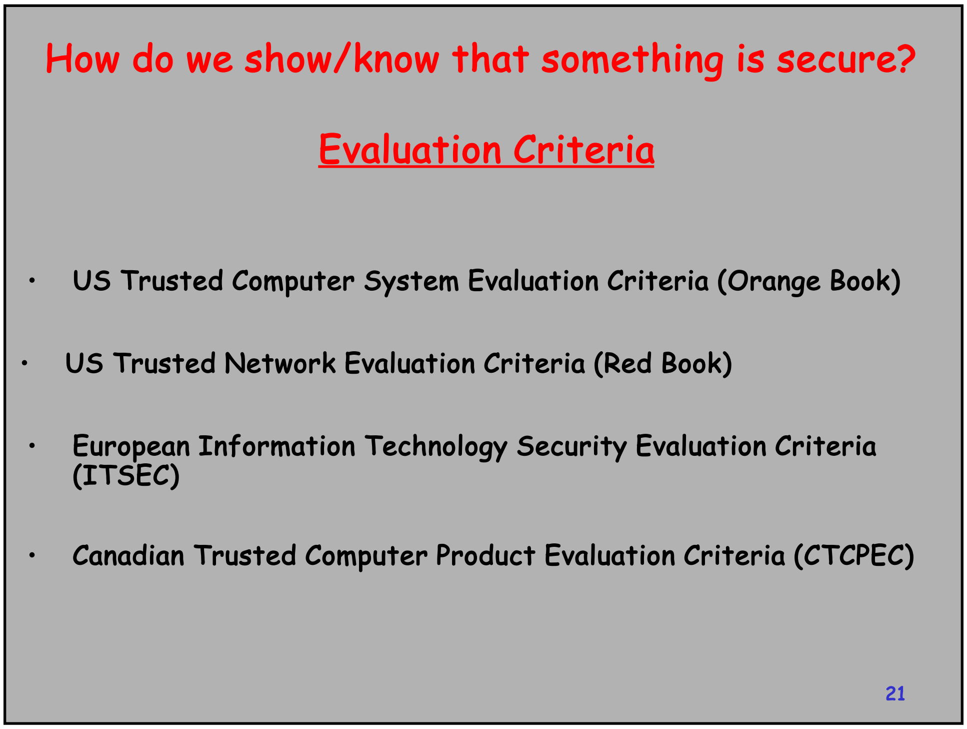


Computer Security

- **Authenticity:** prevention of unauthorized modification of information's origin
- **Accountability: guaranteed logging**, prevention of modification of activity history (logs) and ability to unambiguously attribute actions to entities
- **Reliability:** continued operation after accidental or malicious failures
- **Safety:** limiting impact of failures on the physical environment

How do we evaluate these features?

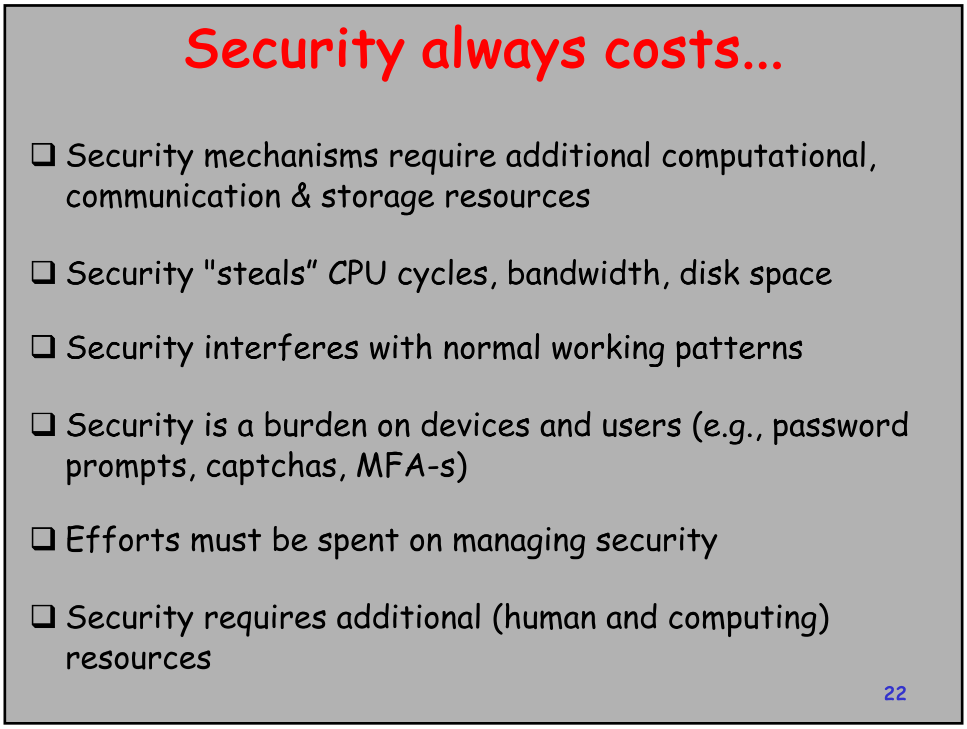
20



How do we show/know that something is secure?

Evaluation Criteria

- US Trusted Computer System Evaluation Criteria (Orange Book)
- US Trusted Network Evaluation Criteria (Red Book)
- European Information Technology Security Evaluation Criteria (ITSEC)
- Canadian Trusted Computer Product Evaluation Criteria (CTCPEC)

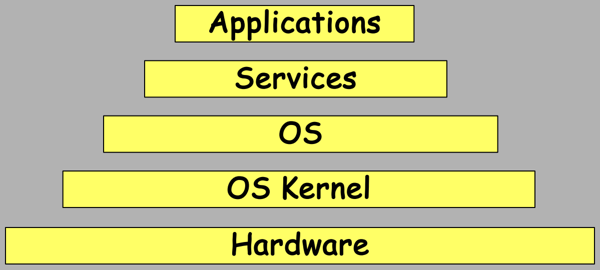


Security always costs…

- ❑ Security mechanisms require additional computational, communication & storage resources
- ❑ Security "steals" CPU cycles, bandwidth, disk space
- ❑ Security interferes with normal working patterns
- ❑ Security is a burden on devices and users (e.g., password prompts, captchas, MFA-s)
- ❑ Efforts must be spent on managing security
- ❑ Security requires additional (human and computing) resources

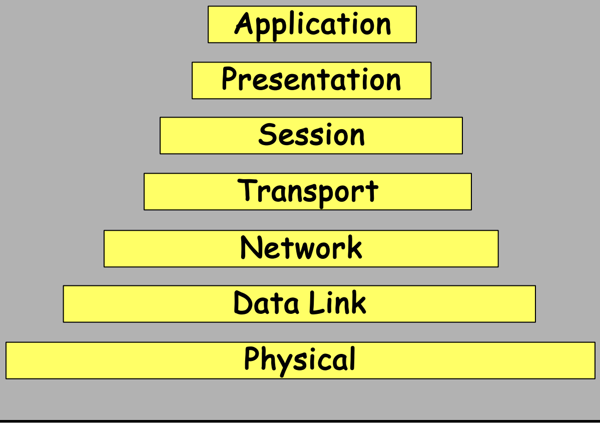
Principles of Computer Security

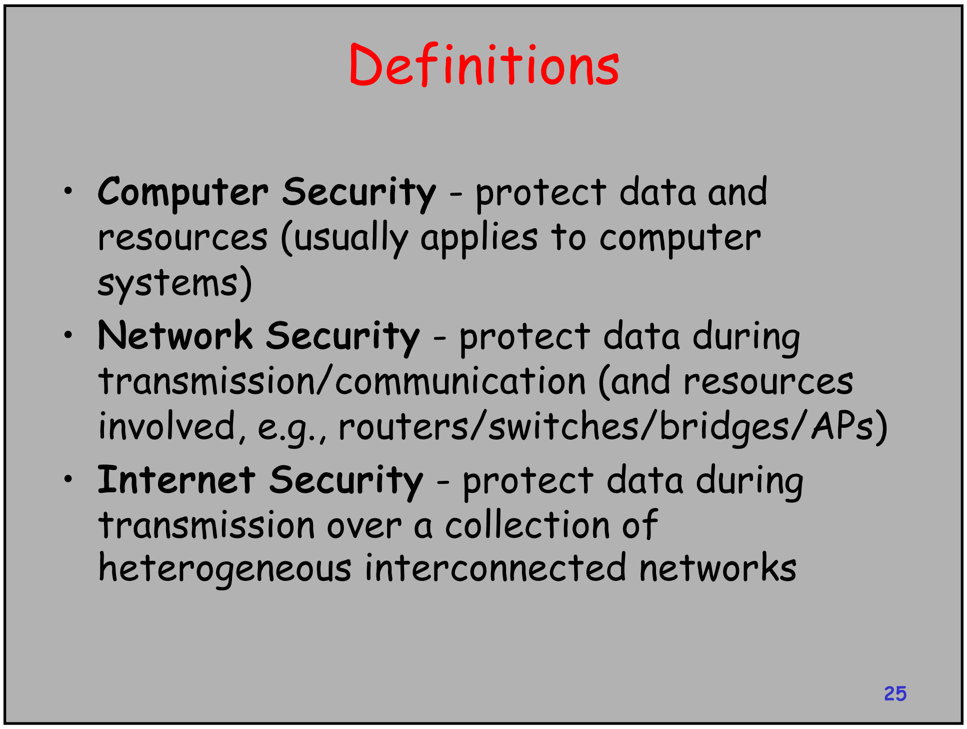
At which layer(s) of the computer system should a security mechanism be placed?



Principles of Network Security

At which layer(s) of the network protocol stack should a security mechanism be placed?



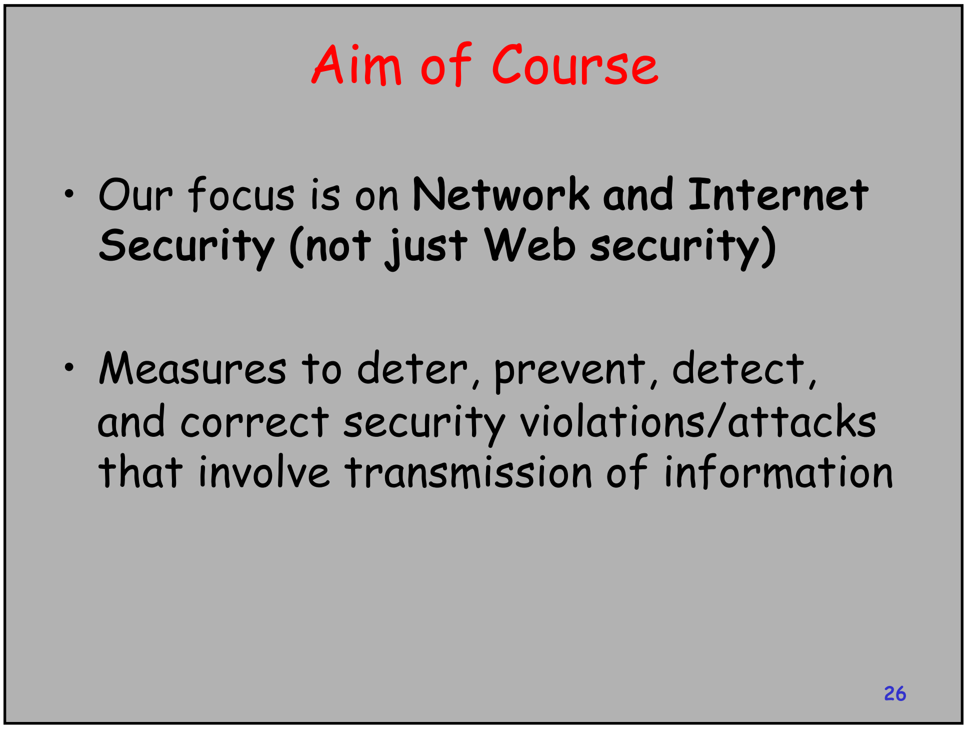


Definitions

- **Computer Security** - protect data and resources (usually applies to computer systems)
- **Network Security** - protect data during transmission/communication (and resources involved, e.g., routers/switches/bridges/APs)
- **Internet Security** - protect data during transmission over a collection of heterogeneous interconnected networks

25

25



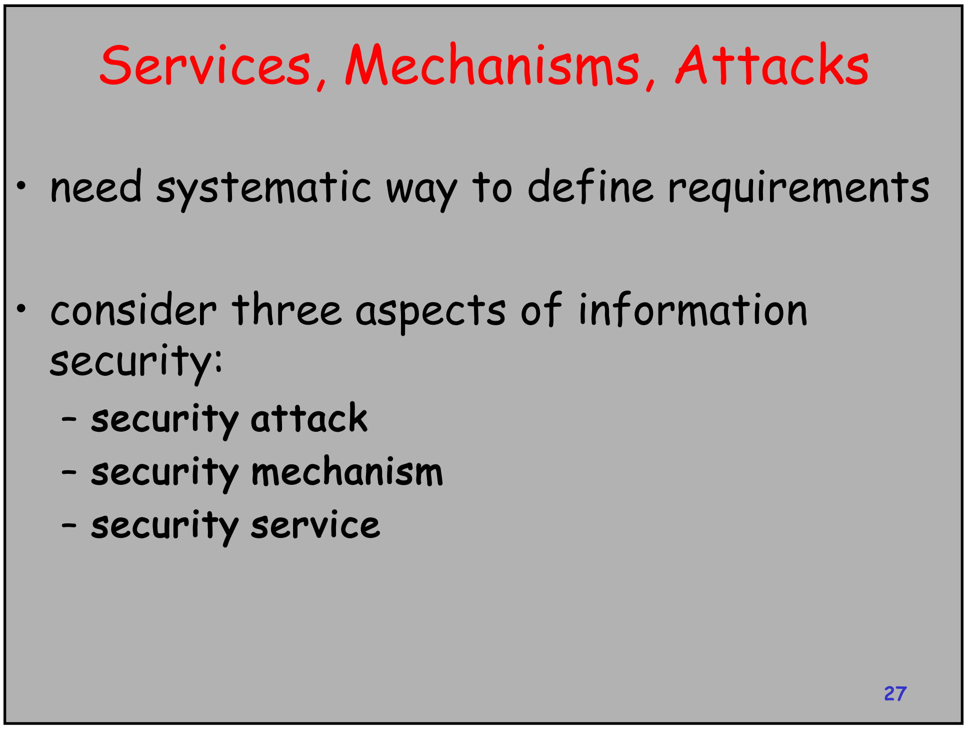
Aim of Course

- Our focus is on **Network and Internet Security (not just Web security)**

- Measures to deter, prevent, detect, and correct security violations/attacks that involve transmission of information

26

26

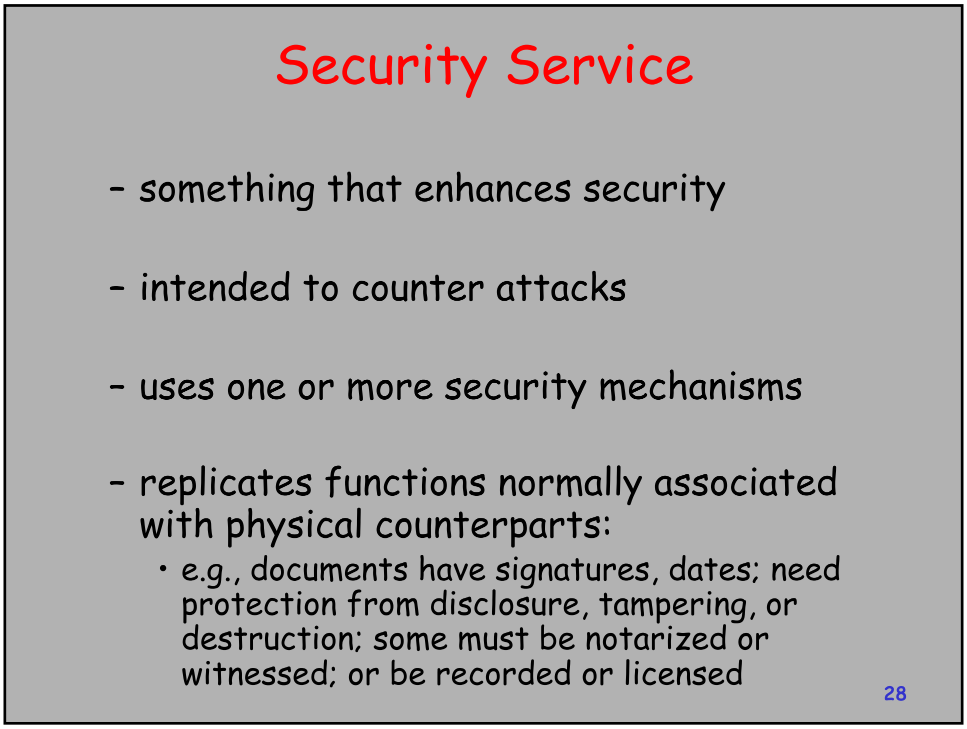


Services, Mechanisms, Attacks

- need systematic way to define requirements
- consider three aspects of information security:
  - **security attack**
  - **security mechanism**
  - **security service**

27

27

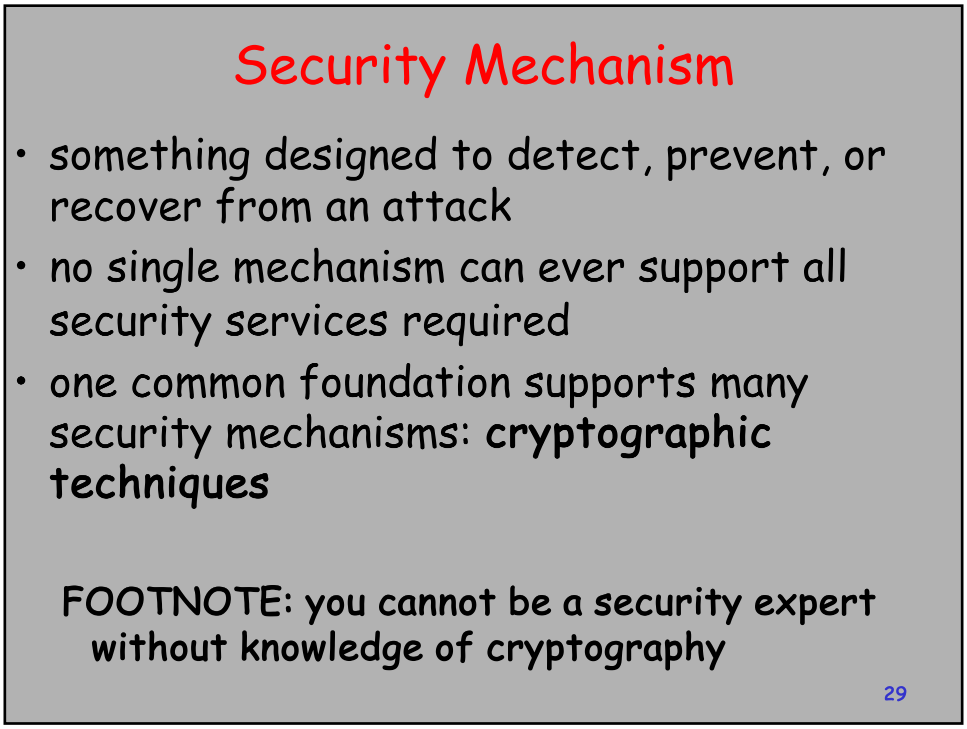


Security Service

- something that enhances security
- intended to counter attacks
- uses one or more security mechanisms
- replicates functions normally associated with physical counterparts:
  - e.g., documents have signatures, dates; need protection from disclosure, tampering, or destruction; some must be notarized or witnessed; or be recorded or licensed

28

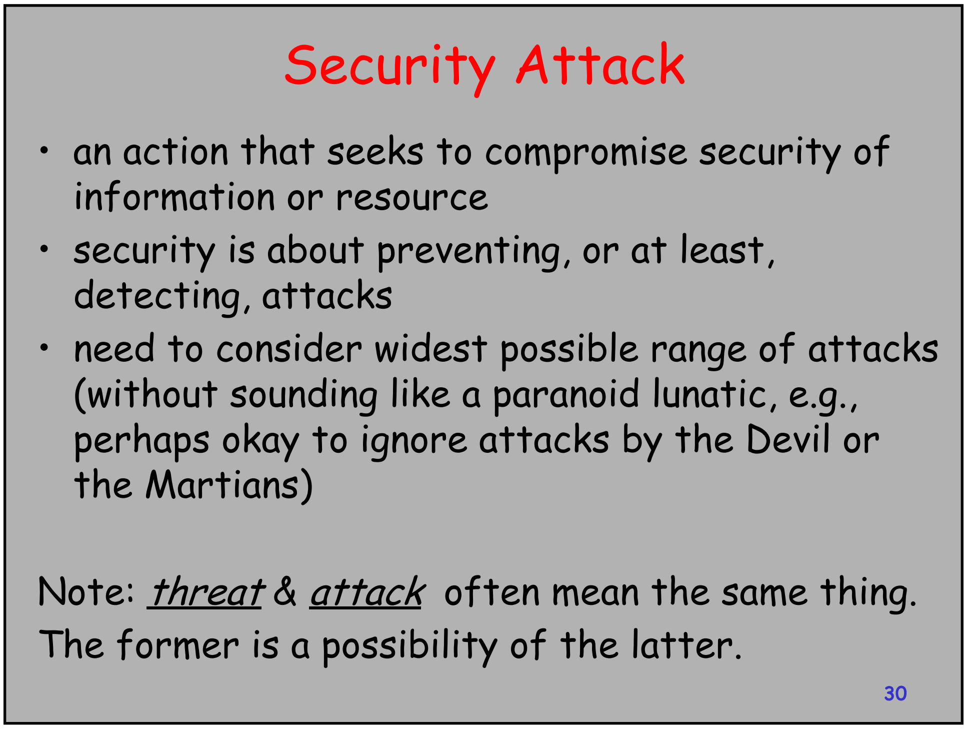
28



Security Mechanism

- something designed to detect, prevent, or recover from an attack
- no single mechanism can ever support all security services required
- one common foundation supports many security mechanisms: **cryptographic techniques**

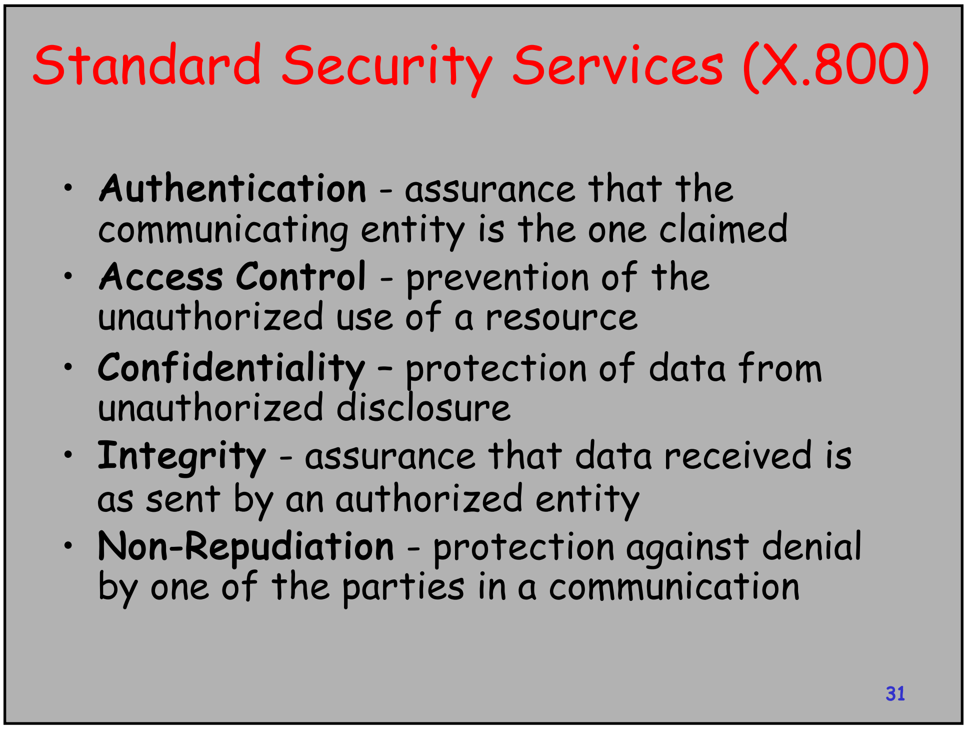
FOOTNOTE: you cannot be a security expert without knowledge of cryptography



Security Attack

- an action that seeks to compromise security of information or resource
- security is about preventing, or at least, detecting, attacks
- need to consider widest possible range of attacks (without sounding like a paranoid lunatic, e.g., perhaps okay to ignore attacks by the Devil or the Martians)

Note: *threat* & *attack* often mean the same thing. The former is a possibility of the latter.

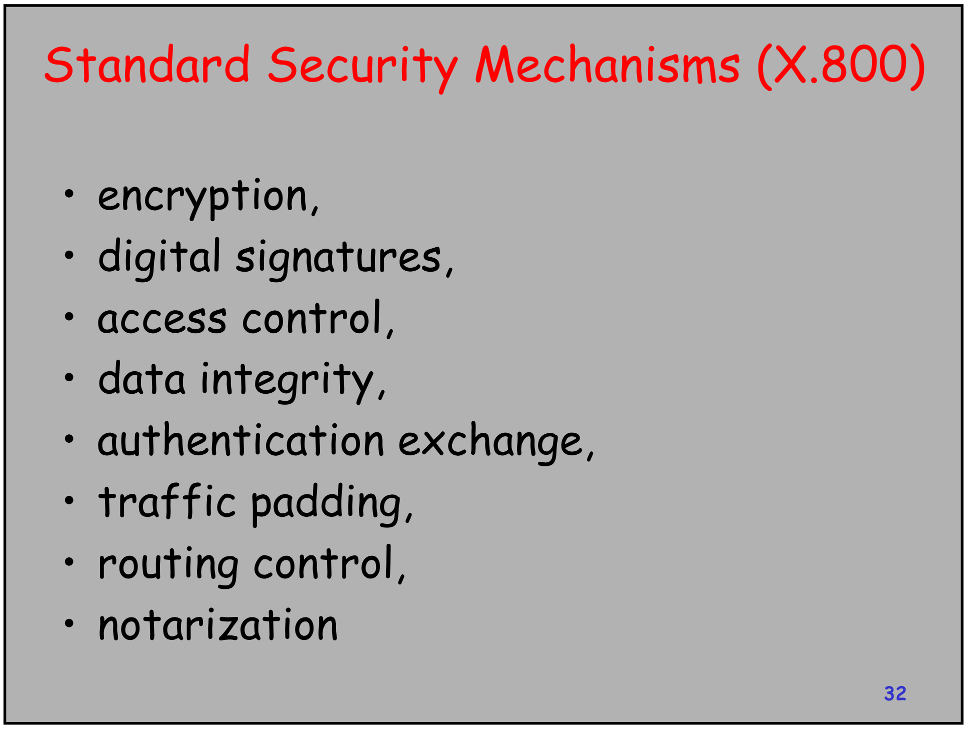


Standard Security Services (X.800)

- **Authentication** - assurance that the communicating entity is the one claimed
- **Access Control** - prevention of the unauthorized use of a resource
- **Confidentiality** – protection of data from unauthorized disclosure
- **Integrity** - assurance that data received is as sent by an authorized entity
- **Non-Repudiation** - protection against denial by one of the parties in a communication

31

31

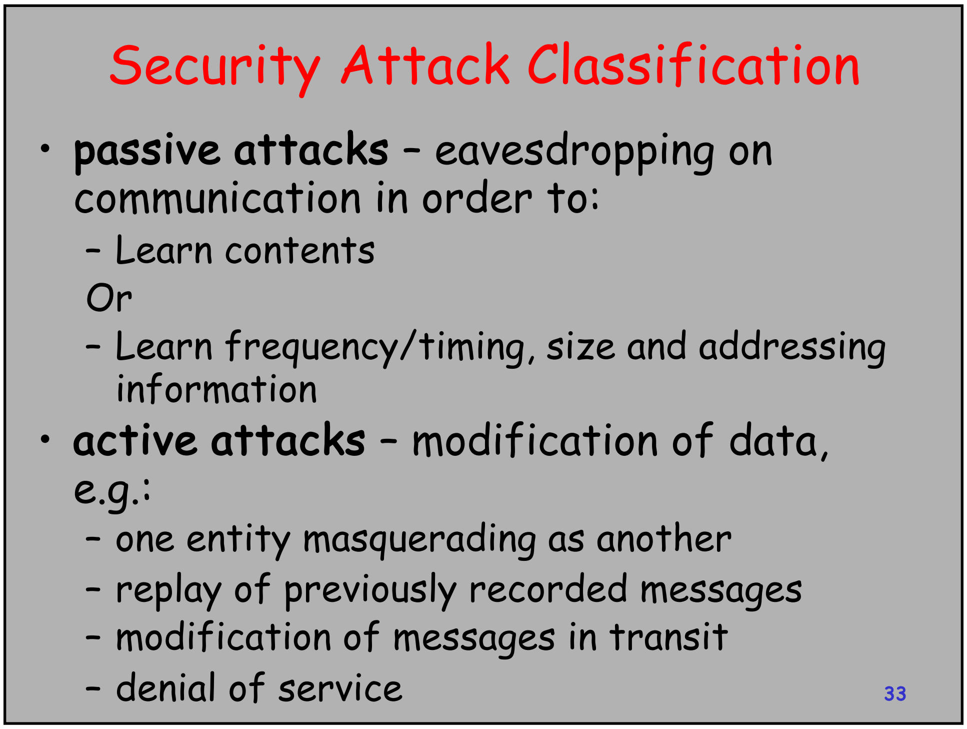


Standard Security Mechanisms (X.800)

- encryption,
- digital signatures,
- access control,
- data integrity,
- authentication exchange,
- traffic padding,
- routing control,
- notarization

32

32

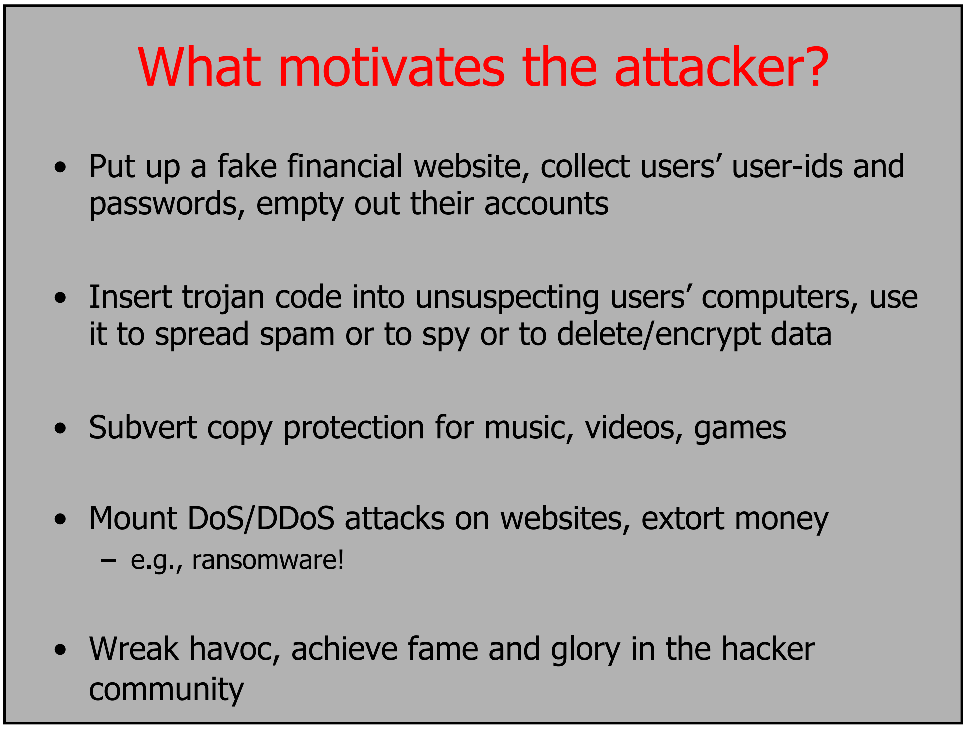


Security Attack Classification

- **passive attacks** – eavesdropping on communication in order to:
  - Learn contents

  Or
  - Learn frequency/timing, size and addressing information
- **active attacks** – modification of data, e.g.:
  - one entity masquerading as another
  - replay of previously recorded messages
  - modification of messages in transit
  - denial of service

33



What motivates the attacker?

- Put up a fake financial website, collect users' user-ids and passwords, empty out their accounts
- Insert trojan code into unsuspecting users' computers, use it to spread spam or to spy or to delete/encrypt data
- Subvert copy protection for music, videos, games
- Mount DoS/DDoS attacks on websites, extort money
  - e.g., ransomware!
- Wreak havoc, achieve fame and glory in the hacker community

35

Marketplace for Vulnerabilities

- Option 1: bug bounty programs
  - Google: <$300K per serious bug
  - META: <$45K -- hw, <$145K -- account takeover
  - Microsoft: up to $150K per bug
  - Pwn2Own competition: $10-15K
- Option 2: vulnerability brokers
  - ZDI, iDefense: $2-150K
- Option 3: gray and black markets
  - Up to $100-500K reported (hard to verify)
  - A zero-day against iOS sold for $500K (allegedly)

36

it's just another kind of business…

- Several companies specialize in finding and selling exploits
  - ReVuln, Vupen, Netragard, Exodus Intelligence
  - The average flaw sells for $35-160K
  - $100K+ annual subscription fees
- Nation-state buyers
  - "Israel, Britain, Russia, India and Brazil are some of the biggest spenders. North Korea is on the market, as are some Middle Eastern intelligence services. Countries in the Asian Pacific, including Malaysia and Singapore, are buying, too" -- NY Times (Jul 2013)

37

Marketplace for Stolen Data

[Dell SecureWorks]

- Single credit card number: $4-15
- Single card with magnetic track data: $12-30
- “Fullz”: $25-40
  - Full name, address, phone, email addresses (with passwords), date of birth, SSN, bank account and routing numbers, online banking credentials, credit cards with magnetic track data and PINs
- Online credentials for a bank account with $70-150K balance: under $300

Prices drop over time, indicating supply glut

38

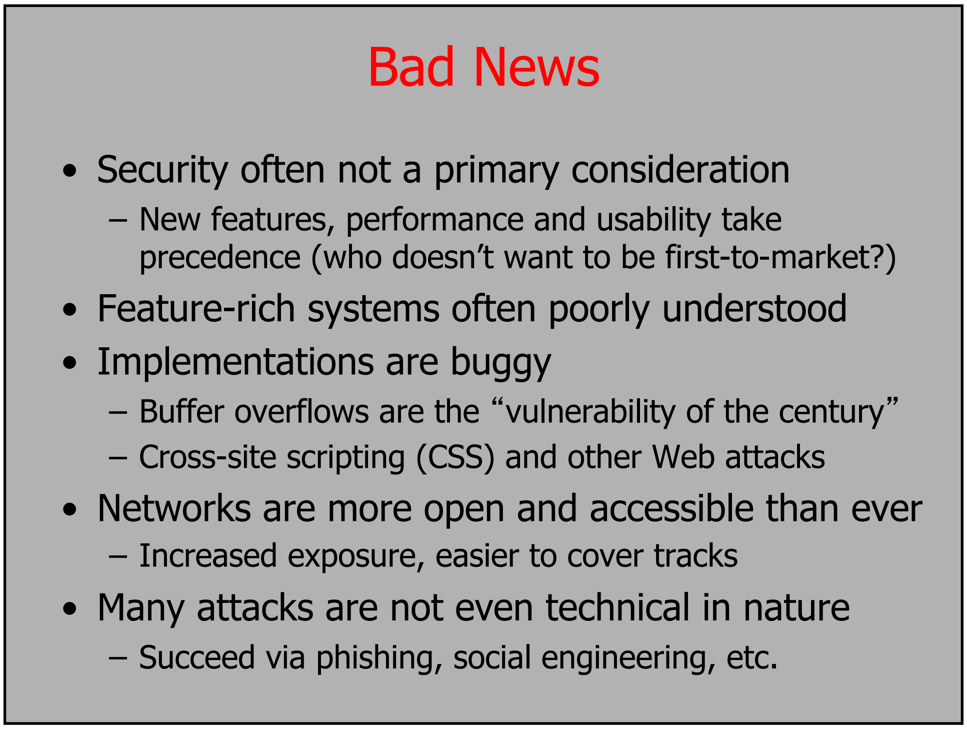
Marketplace for Victims

[Trend Micro, “Russian Underground 101”]

- Pay-per-install on compromised machines
  - US: $100-150 / 1,000 downloads, “global mix”: $12-15
  - Can be used to send spam, stage DoS attacks, perform click fraud, host scam websites
- Botnets for rent
  - DDoS: $10/hour or $150/week
  - Spam: from $10/1,000,000 emails

- Tools and services
  - Basic Trojans ($3-10), Windows rootkits ($300), email, SMS, ICQ spamming tools ($30-50), botnet setup and support ($200/month, etc.)

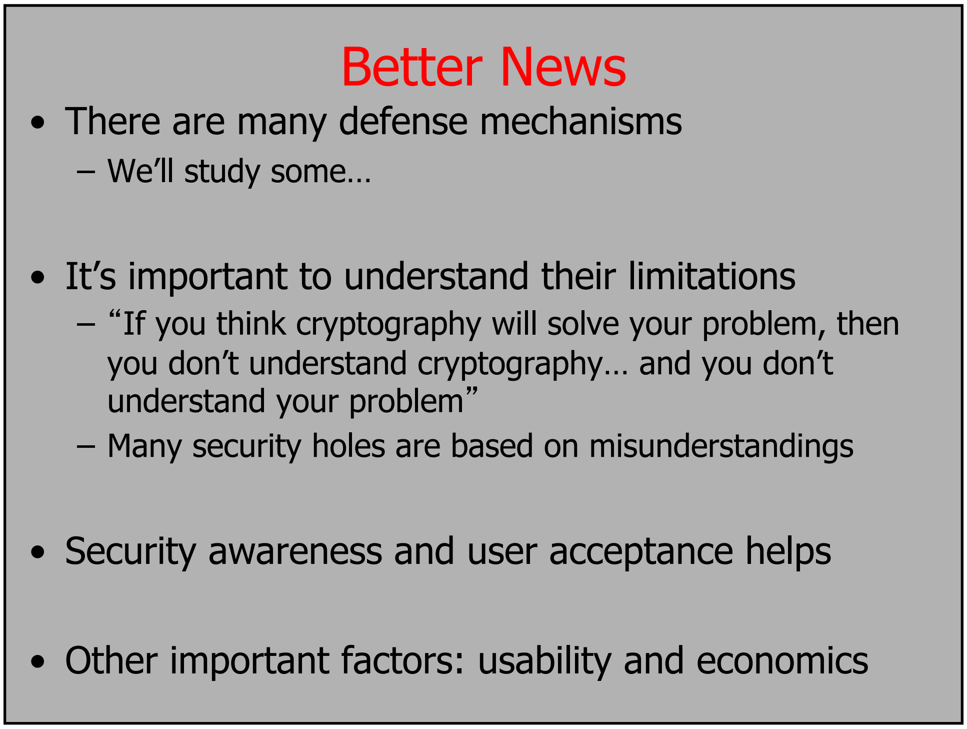
39



Bad News

- Security often not a primary consideration
  - New features, performance and usability take precedence (who doesn’t want to be first-to-market?)
- Feature-rich systems often poorly understood
- Implementations are buggy
  - Buffer overflows are the “vulnerability of the century”
  - Cross-site scripting (CSS) and other Web attacks
- Networks are more open and accessible than ever
  - Increased exposure, easier to cover tracks
- Many attacks are not even technical in nature
  - Succeed via phishing, social engineering, etc.

40

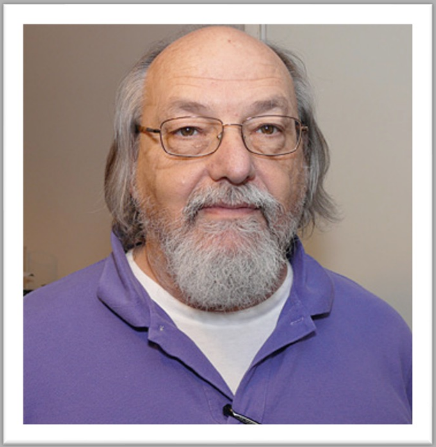


Better News

- There are many defense mechanisms
  - We’ll study some…
- It’s important to understand their limitations
  - “If you think cryptography will solve your problem, then you don’t understand cryptography… and you don’t understand your problem”
  - Many security holes are based on misunderstandings
- Security awareness and user acceptance helps
- Other important factors: usability and economics

41

Ken Thompson

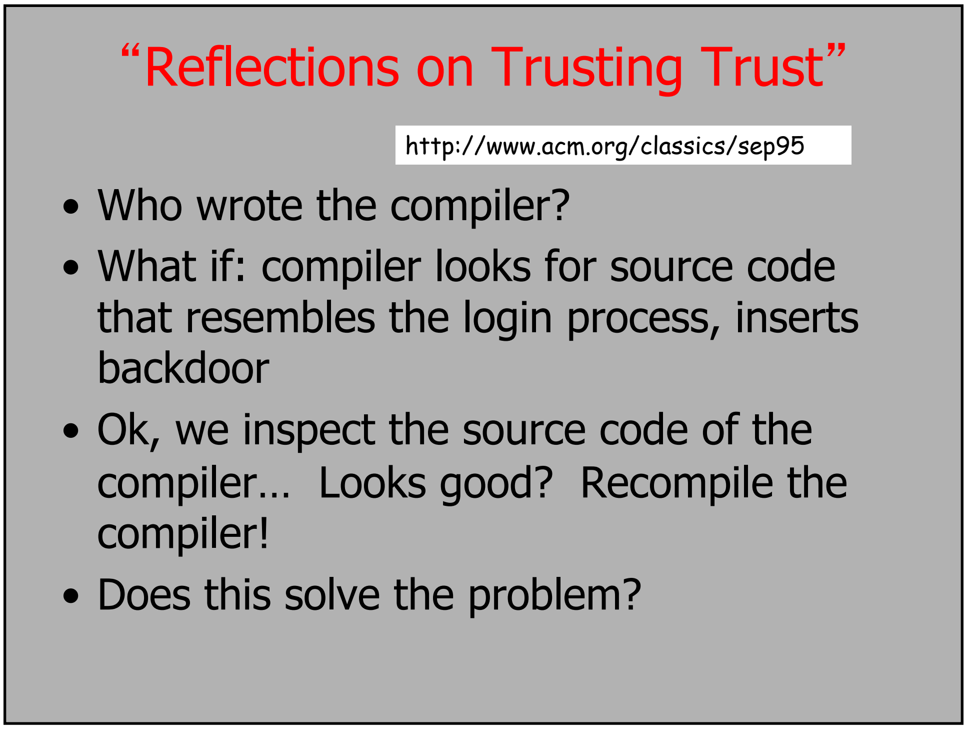


Unix, C co-designer,
1983 ACM Turing Award Recipient

“Reflections on Trusting Trust”

http://www.acm.org/classics/sep95

- What code can we trust?
- Consider "login" or "su" in Unix
  - Is Linux Ubuntu binary reliable? RedHat? Android?
  - Does it send your password to someone?
  - Does it have a backdoor for a “special” remote user?
- Can't trust the binary, so: check source code or write your own, then recompile
- Does this solve the problem?

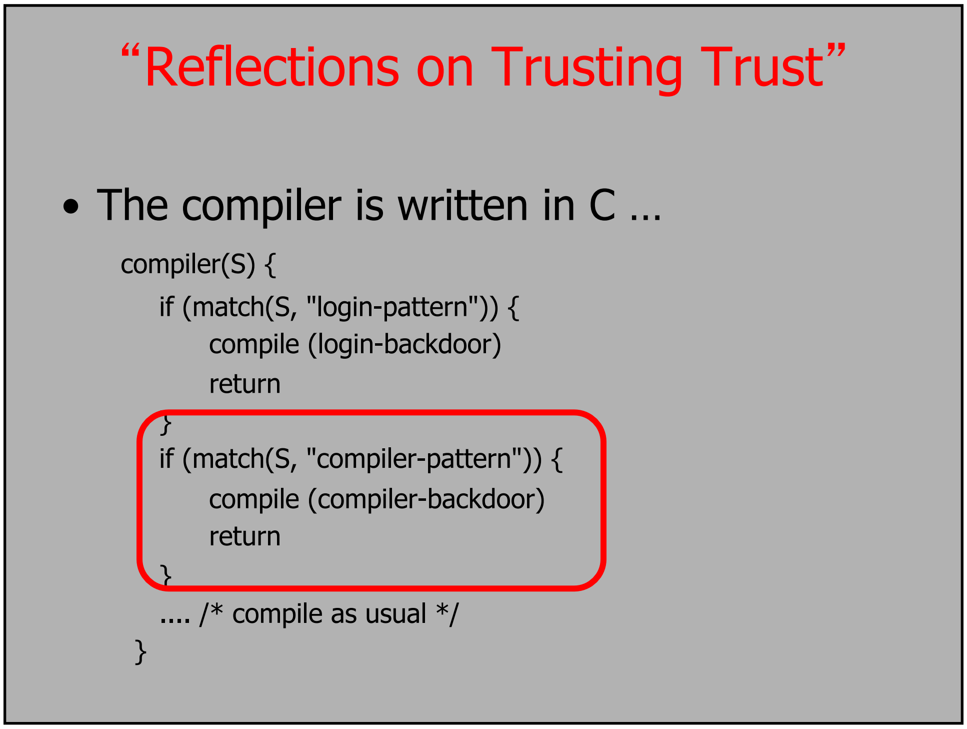


“Reflections on Trusting Trust”

http://www.acm.org/classics/sep95

- Who wrote the compiler?
- What if: compiler looks for source code that resembles the login process, inserts backdoor
- Ok, we inspect the source code of the compiler… Looks good? Recompile the compiler!
- Does this solve the problem?

44



“Reflections on Trusting Trust”

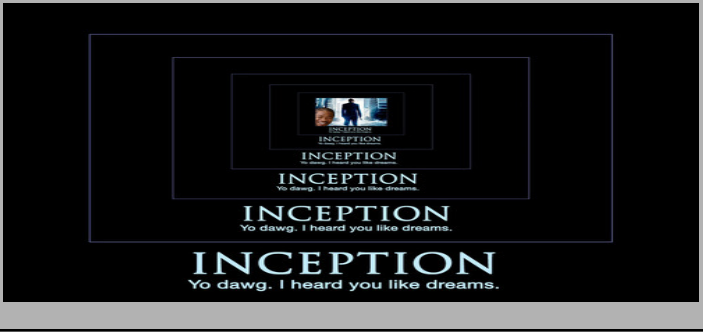
- The compiler is written in C …

```
compiler(S) {
    if (match(S, "login-pattern")) {
        compile (login-backdoor)
        return
    }
    if (match(S, "compiler-pattern")) {
        compile (compiler-backdoor)
        return
    }
    .... /* compile as usual */
  }
```

45

“Reflections on Trusting Trust”

“The moral is obvious. You can't trust code that you did not totally create yourself. (Especially code from companies that employ people like me.)”

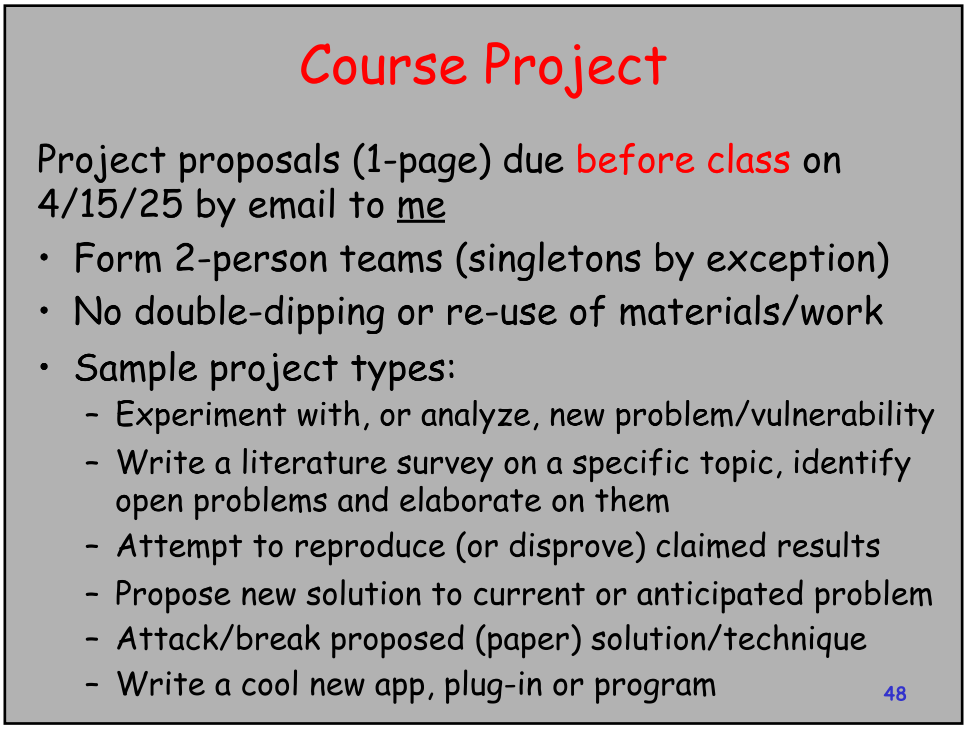


Summary

- We have considered:
  - some basic security definitions
  - security services, mechanisms, attacks
  - models for network security

WHAT YOU NEED TO DO within a week from today:

- See this directory (CS 134 → undergrad course):
  https://drive.google.com/drive/folders/1zaR_JqgqWxboJ1C4r6Js3rZ27ezXwGje?usp=sharing
- Familiarize yourself with that material
- If you can't dig it, drop this course…
- Watch this short video:
  https://www.youtube.com/watch?v=sInf-5i21q8
  Interview with Nicole Perlroth, author of:
  This Is How They Tell Me the World Ends -The Cyberweapons Arms Race
- Team up and think about project topics



Course Project

Project proposals (1-page) due before class on 4/15/25 by email to me

- Form 2-person teams (singletons by exception)
- No double-dipping or re-use of materials/work
- Sample project types:
  - Experiment with, or analyze, new problem/vulnerability
  - Write a literature survey on a specific topic, identify open problems and elaborate on them
  - Attempt to reproduce (or disprove) claimed results
  - Propose new solution to current or anticipated problem
  - Attack/break proposed (paper) solution/technique
  - Write a cool new app, plug-in or program