PennState

CSE 597
Security of Emerging Technologies
Module: Introduction

Prof. Syed Rafiul Hussain
Systems and Network Security (SyNSec) Research Group
Department of Computer Science and Engineering
The Pennsylvania State University

About Me

- Education:
  - 2013-2018: Ph.D. from Purdue University in CS
  - 2019-2020: Postdoc at Purdue University
  - 2020: Assistant Professor at Penn State
- Research Interests: Network and Systems Security
  - Cellular Network (4G and 5G) Security
  - Formal Verification
  - Program Analysis
  - Fuzzing and Software Security

Most of my research work is grounded on
formal verification, program analysis, software testing, and cryptography

Some bedtime stories …


ZDNet VIDEOS

Heartbleed: Serious OpenSSL zero day vulnerability revealed

A new OpenSSL vulnerability has shown up and some companies are annoyed that the bug was revealed l
patches could be delivered for it. Updated April 8.

By Steven J. Vaughan-Nichols for Networking | April 7, 2014 --
22:34 GMT (15:34 PDT) | Topic: Security


ZDNet

Spectre and Meltdown: Insecurity at the heart of modern CPU design

y-discovered flaws in many processors threaten performance hits and continued security headaches. Here's how
work, how they got there, and what they mean for the future.

By Rupert Goodwins | January 9, 2018 -- 11:05 GMT (03:05 PST) |
Topic: Security


Cybercriminals Target Hospitals with SamSam Ransomware Attacks

Cybercriminals increased their SamSam ransomware

WIRED BACKCHANNEL BUSINESS CULTURE GEAR IDEAS SCIENCE SECURITY

DAN GOODIN, ARS TECHNICA SECURITY 08.23.2020 09:00 AM

A New Botnet Is Covertly Targeting Millions of Servers

FritzFrog has been used to try and infiltrate government agencies, banks, telecom companies, and universities across the


TC

New flaws in 4G, 5G allow attackers to intercept calls and track phone locations

Zack Whittaker @zackwhittaker / 11:39 am EST • February 24, 2019


The Washington Post
Democracy Dies in Darkness

‘I’m in your baby’s room’: A hacker took over a baby monitor and broadcast threats, parents say

What is Information (Computer) Security?

- Security = Sustain desirable properties under intelligent adversaries
- Make the above precise requires making the following two precise

- Desirable properties
  - Understand what properties are needed.
- Intelligent adversaries
  - Needs to understand/model adversaries
  - Always think about adversaries.

What is Security

- Operational Definition of Security:

> Security is the application and enforcement of policies through mechanisms over data and resources

- **Policies** specify what we want to enforce (e.g., only Alice should read file f)
  - Common goals: confidentiality, integrity, availability
- **Mechanisms** specify how we enforce the policy (i.e., an implementation of a policy, encryption)
  - Goal has nothing to say about mechanism

Security Principles

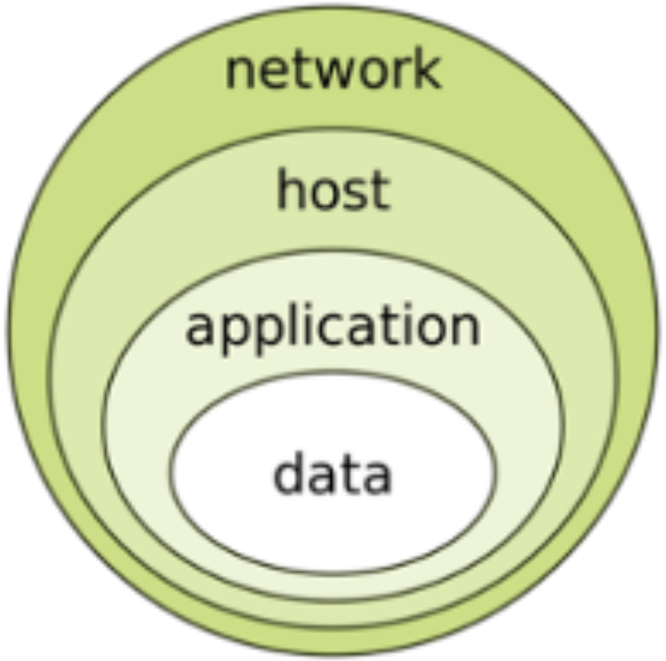
- Confidentiality
  - Only those who are authorized to know can know
  - an attacker cannot recover protected data
- Integrity
  - (Data) Only modified by authorized parties and in permitted ways
  - (Systems) do things as expected and free from unauthorized modifications
  - an attacker cannot modify protected data/systems
- Availability
  - those authorized to access can get access
  - an attacker cannot stop/hinder computation/authorized access
- Accountability/non-repudiation (fourth fundamental concept)

Why Do Computer Attacks

- Who are the attackers?
  - Bored teenagers, criminals, organized crime organizations, rogue (or other) states, industrial espionage, angry employees, …
- Why do they do it?
  - fun,
  - fame,
  - profit, …
- Computer systems are where the moneys are
  - Political/military objectives

Security of a system

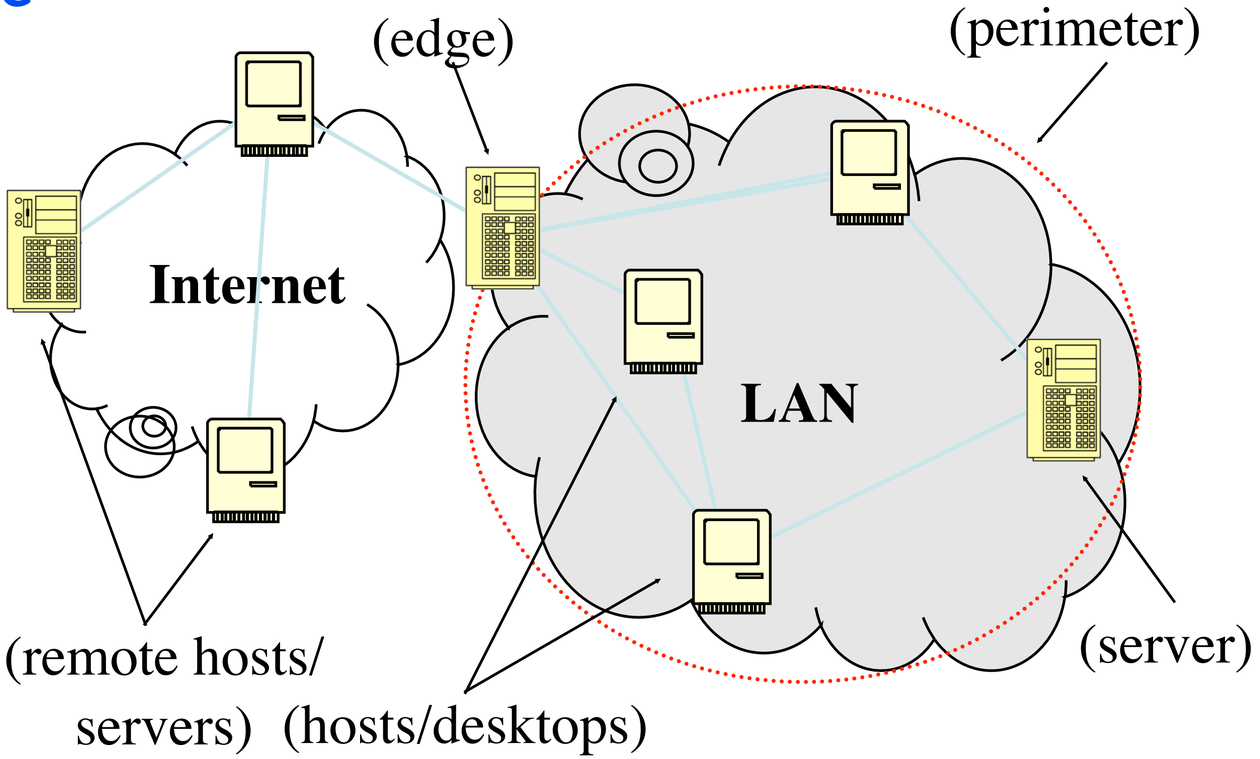
- Security applied to computer systems
  - Hardware, software, network, and computing power
  - Protection against theft, damage, misuse, and disruption of the services they provide



Security of Complex Systems

Security mechanisms and policies have been implemented at several system layers (app, OS, VM, network)

Are we now secure?



Current Security Problems

Most current security problems are based on the failure of people to deploy hosts and networks securely.

Botnets

Rootkits

Web attacks: XSS, SQL Inject, …

Worms (Conficker, Stuxnet)

Password Guessing

Buffer Overflows

Arbitrary App Flaws

Side-Channel Attacks

Tracking Users

Intercepting Phone calls

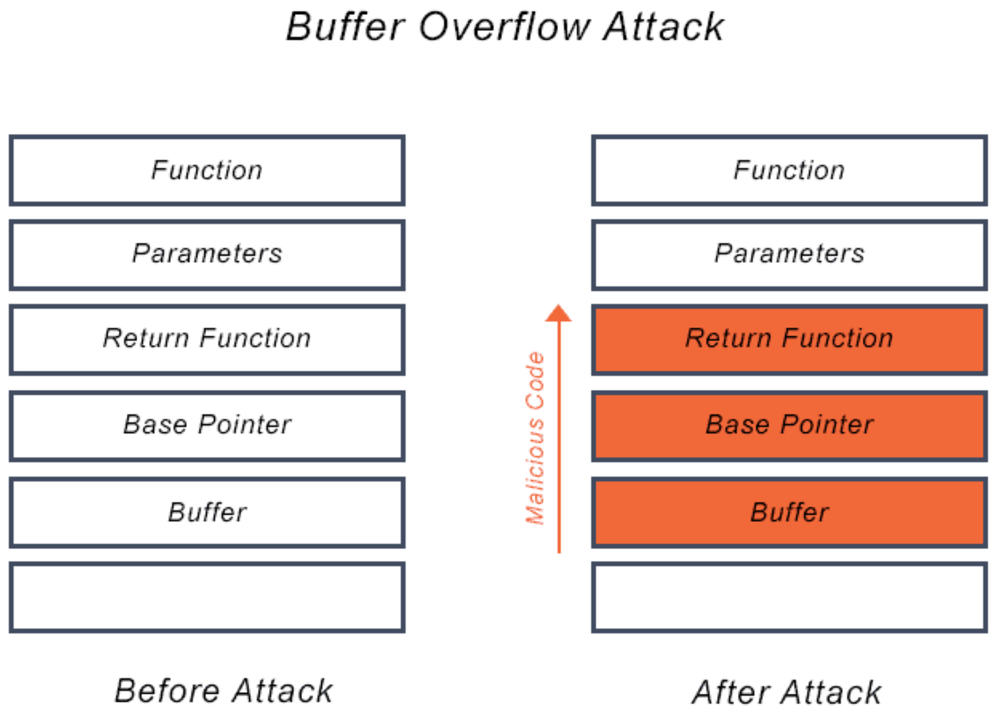
Security State

SANS Top Security Risks

http://www.sans.org/top-cyber-security-risks/

- Client-side software is unpatched (apps patched slower)
- Web servers are vulnerable (XSS are 80%)
- Application vulnerabilities exceed OS vulnerabilities
- Attacks on Mac systems (QuickTime)
- US is the major attack target (30:1)
- Still buffer (and heap) overflows
- DNS Cache Po

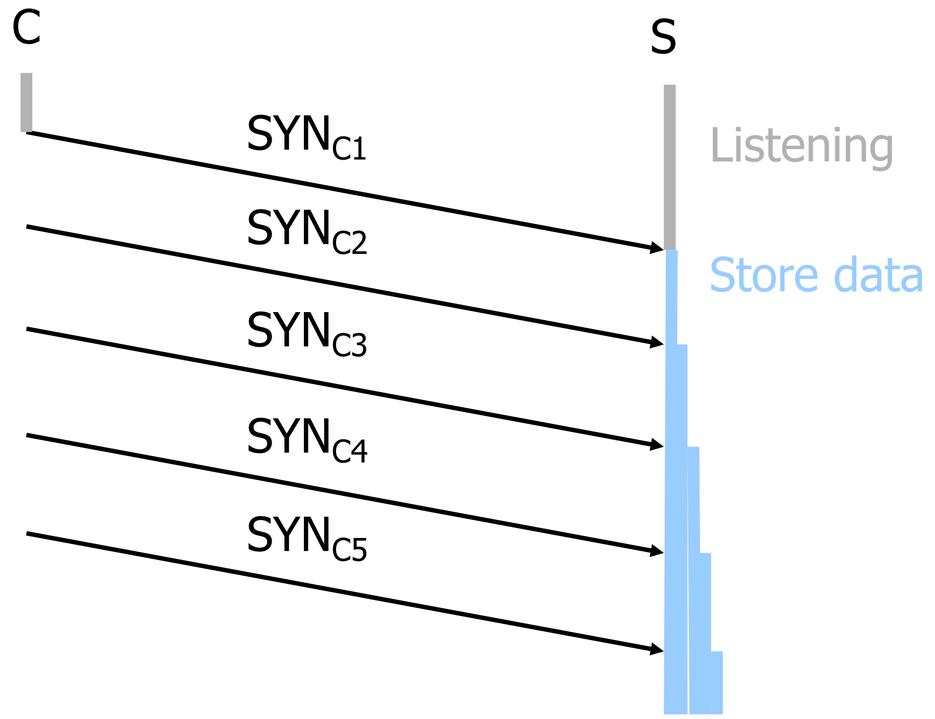
Example Attack 1



- Overwrite the return address with the address of the attacker code.

Example Attack 2

- TCP SYN Flooding and DDoS Attack
  - Attacker sends many connection requests
    - Spoofed source addresses
  - Victim allocates resources for each request
    - Connection requests exist until timeout
    - Old implementations have a small and fixed bound on half-open connections
  - Resources exhausted => requests rejected
  - No more effective than other channel capacity-based attack today



Example Attack 3

- Aim: Get a client to run an attackers' code at higher privilege (Privilege Escalation)
- Attack:
  - Attacker places content on trusted site
  - Client downloads content and that content attacks an unpatched client program (e.g., media player)
  - Attacker can run as client user
  - Install reverse shell backdoor (outbound HTTPS)
  - Download local privilege escalation program (again unpatched client code)
  - Attack other machines – Windows domain controller

Security Mythology

- Claim: *All these problems were solved*

- Is this claim true?
- Why not just use it?
- What is necessary?
- By whom?
- Can we make it happen?

Why is Security Hard?

PennState

Things can go wrong in multiple fronts

- Problems with policy
  - Examples:
  - Reset password by answering security questions
  - Reset password by providing last 4 digits of credit card numbers

Why is Security Hard?

Things can go wrong in multiple fronts

- Problems with assumptions
  - Examples:
  - Human factors – the weakest link in phishing attacks
  - Hardware is trustworthy
  - Attackers' computational power

Why is Security Hard?

PennState

Things can go wrong in multiple fronts

- Problems with mechanisms
  - Bugs/vulnerabilities in implementations
  - Examples:
    - Impersonation attacks
      - Apple's "goto" fail
  - Information leakage
    - OpenSSL's Heartbleed bug
  - Unauthorized access
    - Unlimited password-guessing allowed for one iCloud API

Answer?

- Analysis Tools for systems, networks and programs
  - 1980s – 90s: formal verification methods
  - 2000s: Bug finding
  - 2010s: tools to find and fix security bugs?
  - Beyond??
- Problem: what bugs should be discovered?
- Problem: soundness and completeness
- Problem: how should analysis impact software development and system deployment?

Who Has the Role?

- Programmers (may be multiple groups)
- OS Distributors
- Administrators
- Users
- Network Service Providers
- Content Providers
- **Challenge:** Must consider the balance between function and security

This course …

- Is a software course that teaches principles and techniques for analyzing security properties
  - Lots of techniques have been developed, but we need to figure out how to use/extend them to improve systems security
  - Topics:
    - What should "secure" mean in networking and systems?
    - How to find violations of security in programs, networks and systems?
    - How to fix such violations of security automatically?
    - How to make such techniques tractable and practical?
    - How to design high-assurance or formally verified defense?

Bugs

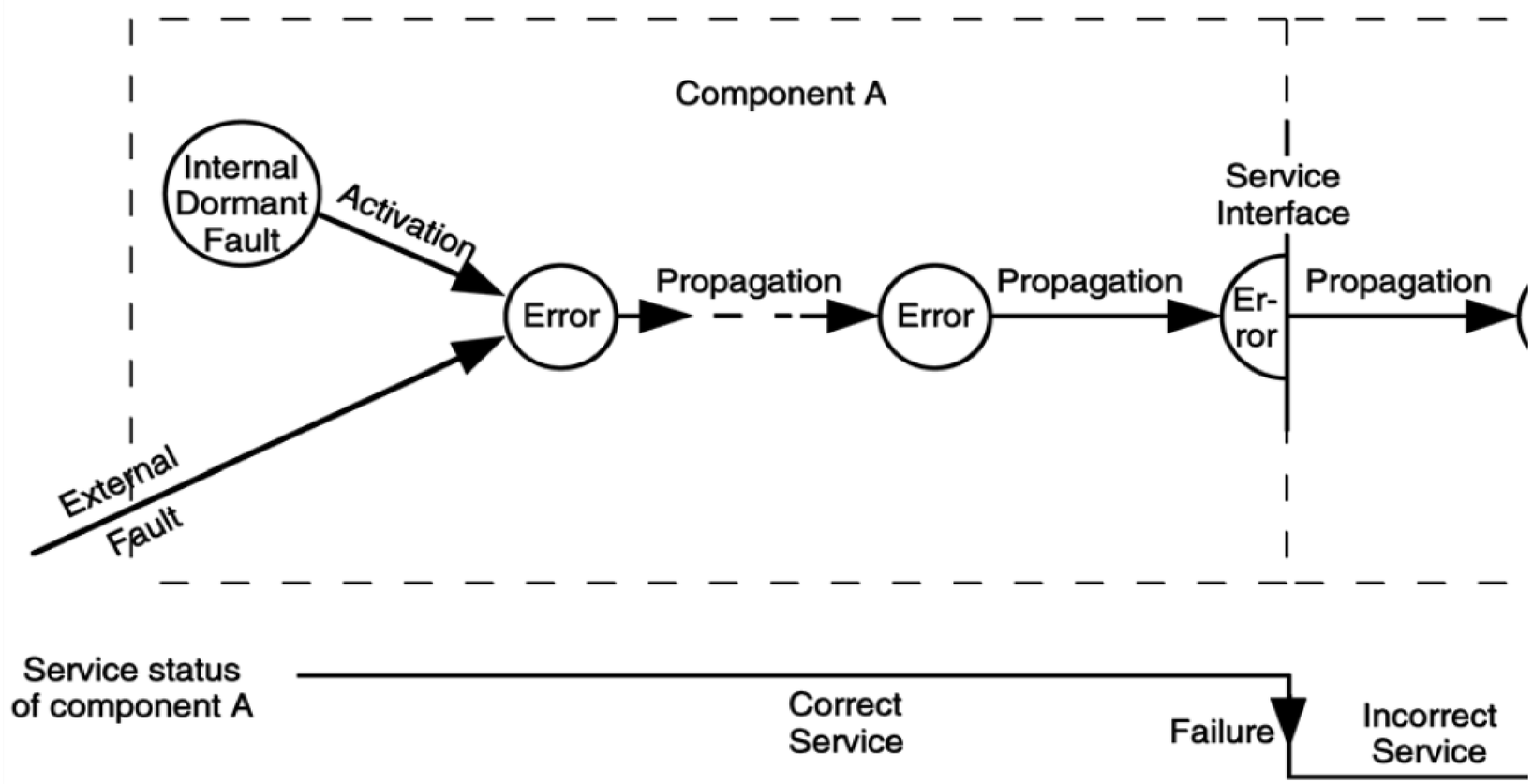
A software bug is an error, flaw, failure, defect or fault in a computer program or system that causes it to produce an incorrect or unexpected result, or to behave in unintended ways. [Wikipedia]

[1] Avizienis *et al*., Basic Concepts and Taxonomy of Dependable and Secure Computing, IEEE TDSC, 1(1), 2014 (https://www.nasa.gov/pdf/636745main_day_3-algirdas_avizienis.pdf)

Faults, Errors & Failures

- Faults
  - Physical defects or flaws occurring in some components (hardware or software)
    - E.g., Buffer overflow vulnerability
- Errors
  - Introduce inaccuracy in system's state due to some faults
    - E.g., Memory corruption
- Failures
  - Not delivering the correct/expected service
    - E.g., System crash
-

Faults, Errors & Failures



- A fault is active when it causes an error, otherwise it is dormant
- Many errors do not reach the system's external interface, as a result, no failure is observed

Vulnerabilities

A vulnerability is a weakness which allows an attacker to violate the system's security policies. [Wikipedia]

- A vulnerability is exploitable
  - If an exploit (e.g., malicious inputs) exists
- Depend on the goal(s) of the attacker(s)
- Some vulnerabilities are not related to software
  - Human factors can result in, e.g., phishing attacks
- Security bugs
  - Bugs in security enforcing software (e.g., OS kernel, SSL/TLS)
  - Lead to vulnerabilities and leave a system open to attacks

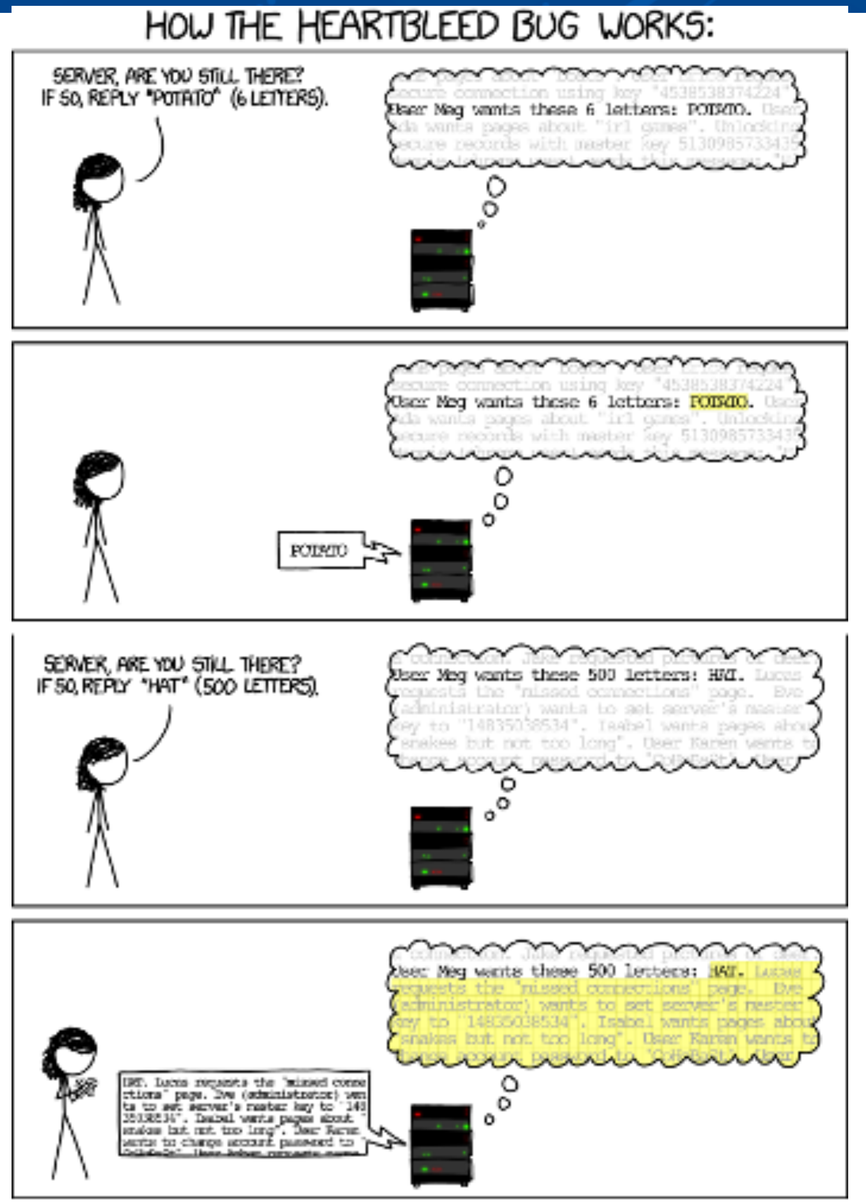
Real World Security Incidents

- Symantec
  - Cyber crime hit the big time in 2016, with high-profile victims and bigger than ever financial rewards
  - discovered more than 430 million new unique pieces of malware in 2015, up 36 percent from the year before.
- Trend Micro
  - predicts a 25% growth in the number of new ransomware families in 2017
  - IoT devices will play a bigger role in DDoS attacks
  - the average payout for a ransomware attack is $722, which could reach up to $70K if an enterprise network is hit.

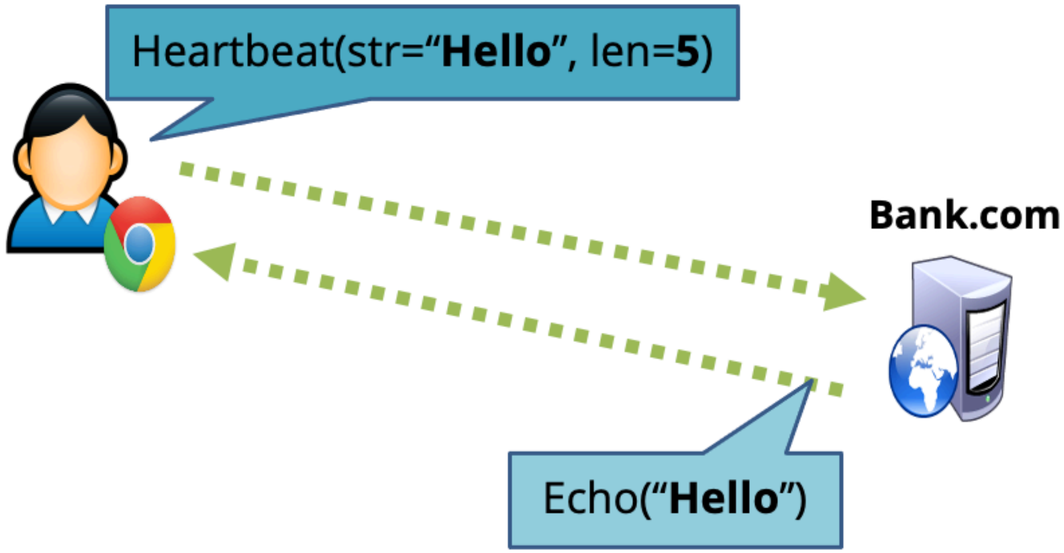
Bug in OpenSSL Hearbeat

- TLS heartbeat extension
  - A **keep-alive feature**
  - One end of the connection sends a payload of arbitrary data to the other end
  - The other end sends back an exact copy of that data to prove everything's OK.

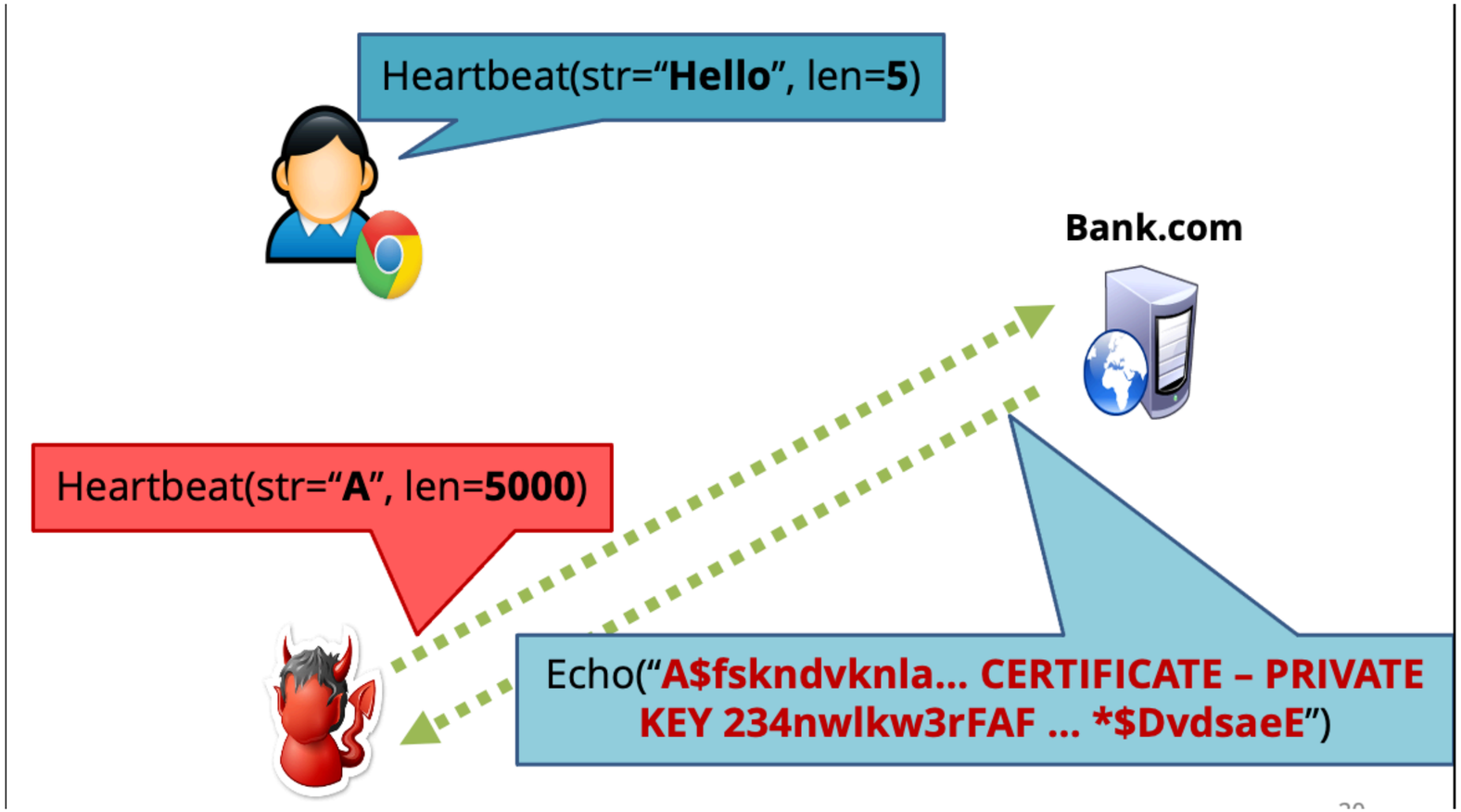
- Bug in OpenSSL
  - Buffer over-read due to missing bounds check
- Reveal in-memory authentication credentials
  - Threat to confidentiality



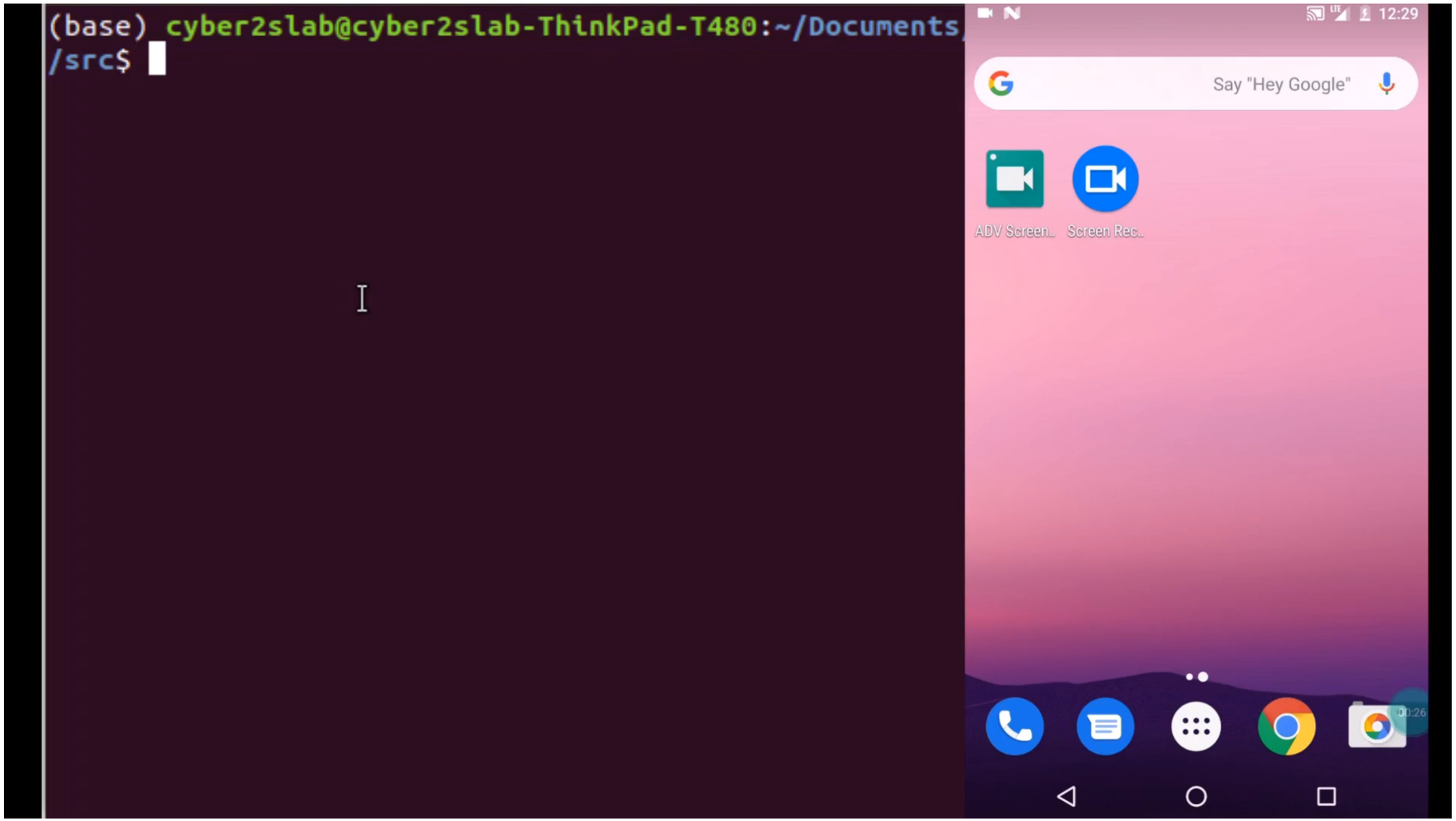
OpenSSL Heartbleed



OpenSSL Heartbleed



Demo of an Attack



Security is Not Absolute

- Is your phone secure?
- What does “secure” mean?

- Security is relative
  - to the kinds of loss one consider
    - security objectives/properties need to be stated
  - to the threats/adversaries under consideration.
    - security is always under certain assumptions

Security is Interesting

- The most interesting/challenging threats to security are posed by human adversaries
  - Security is harder than reliability
- Information security is a self-sustaining field
  - Can work both from attack perspective and from defense perspective
- Security is about benefit/cost tradeoff
  - Thought often the tradeoff analysis is not explicit
- Security is not all technological
  - Humans are often the weakest link

Security is Challenging

- Defense is almost always harder than attack.
- In which ways information security is more difficult than physical security?
  - adversaries can come from anywhere
  - computers enable large-scale automation
  - adversaries can be difficult to identify
  - adversaries can be difficult to punish
  - potential payoff can be much higher
- In which ways information security is easier than physical security?

Background

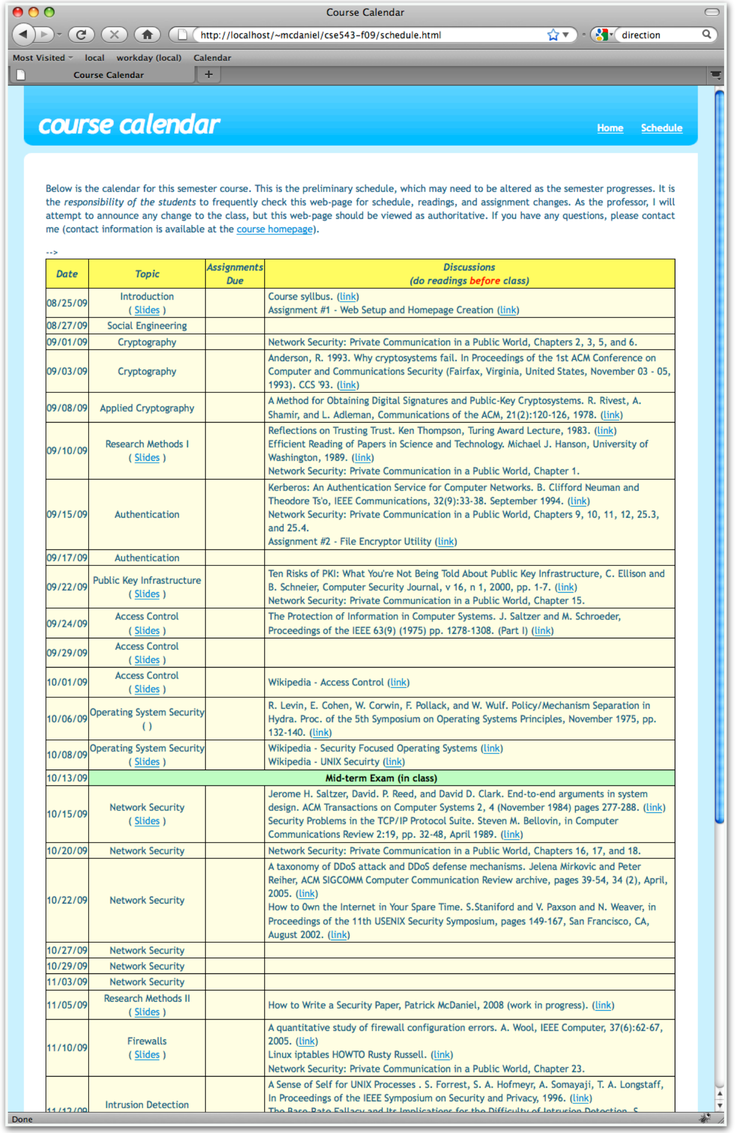
- Required
  - CSE 543 or CSE 443 or Equivalent Course from your undergraduate or MS study
- Expected
  - Operating Systems
  - Networking
  - Basic Security
- Additional
  - Willingness to read
    - We are going to read some papers on security and analysis techniques
  - Willingness to program
    - The project will have coding/implementations.

Course Info

- Website - I am maintaining the course website at
  - https://syed-rafiul-hussain.github.io/index.php/teaching/cse597-s25/
  - The course syllabus can also be found on CANVAS where course related policies are discussed. Changes may apply.
- Office Hours:
  - Tuesday and Thursday (2:00pm-3:00pm)
-

Course Calendar

- The course calendar has all the relevant readings, paper presentations, paper reviews and project milestones
- The calendar page contains electronic links to online papers assigned for course readings.
- *Please check the website frequently for announcements and changes* to the schedule. Students are responsible for any change on the schedule.



Date	Topic	Assignments Due	Discussions (do readings before class)
08/25/09	Introduction (Slides)		Course syllbus. (link) Assignment #1 - Web Setup and Homepage Creation (link)
08/27/09	Social Engineering		
09/01/09	Cryptography		Network Security: Private Communication in a Public World, Chapters 2, 3, 5, and 6.
09/03/09	Cryptography		Anderson, R. 1993. Why cryptosystems fail. In Proceedings of the 1st ACM Conference on Computer and Communications Security (Fairfax, Virginia, United States, November 03 - 05, 1993). CCS '93. (link)
09/08/09	Applied Cryptography		A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. R. Rivest, A. Shamir, and L. Adleman, Communications of the ACM, 21(2):120-126, 1978. (link)
09/10/09	Research Methods I (Slides)		Reflections on Trusting Trust. Ken Thompson, Turing Award Lecture, 1983. (link) Efficient Reading of Papers in Science and Technology. Michael J. Hanson, University of Washington, 1989. (link) Network Security: Private Communication in a Public World, Chapter 1.
09/15/09	Authentication		Kerberos: An Authentication Service for Computer Networks. B. Clifford Neuman and Theodore Ts'o, IEEE Communications, 32(9):33-38. September 1994. (link) Network Security: Private Communication in a Public World, Chapters 9, 10, 11, 12, 25.3, and 25.4. Assignment #2 - File Encryptor Utility (link)
09/17/09	Authentication		
09/22/09	Public Key Infrastructure (Slides)		Ten Risks of PKI: What You're Not Being Told About Public Key Infrastructure, C. Ellison and B. Schneier, Computer Security Journal, v 16, n 1, 2000, pp. 1-7. (link) Network Security: Private Communication in a Public World, Chapter 15.
09/24/09	Access Control (Slides)		The Protection of Information in Computer Systems. J. Saltzer and M. Schroeder, Proceedings of the IEEE 63(9) (1975) pp. 1278-1308. (Part I) (link)
09/29/09	Access Control (Slides)		
10/01/09	Access Control (Slides)		Wikipedia - Access Control (link)
10/06/09	Operating System Security ( )		R. Levin, E. Cohen, W. Corwin, F. Pollack, and W. Wulf. Policy/Mechanism Separation in Hydra. Proc. of the 5th Symposium on Operating Systems Principles, November 1975, pp. 132-140. (link)
10/08/09	Operating System Security (Slides)		Wikipedia - Security Focused Operating Systems (link) Wikipedia - UNIX Secuirty (link)
10/13/09	**Mid-term Exam (in class)**		
10/15/09	Network Security (Slides)		Jerome H. Saltzer, David. P. Reed, and David D. Clark. End-to-end arguments in system design. ACM Transactions on Computer Systems 2, 4 (November 1984) pages 277-288. (link) Security Problems in the TCP/IP Protocol Suite. Steven M. Bellovin, in Computer Communications Review 2:19, pp. 32-48, April 1989. (link)
10/20/09	Network Security		Network Security: Private Communication in a Public World, Chapters 16, 17, and 18.
10/22/09	Network Security		A taxonomy of DDoS attack and DDoS defense mechanisms. Jelena Mirkovic and Peter Reiher, ACM SIGCOMM Computer Communication Review archive, pages 39-54, 34 (2), April, 2005. (link) How to Own the Internet in Your Spare Time. S.Staniford and V. Paxson and N. Weaver, in Proceedings of the 11th USENIX Security Symposium, pages 149-167, San Francisco, CA, August 2002. (link)
10/27/09	Network Security		
10/29/09	Network Security		
11/03/09	Network Security		
11/05/09	Research Methods II (Slides)		How to Write a Security Paper, Patrick McDaniel, 2008 (work in progress). (link)
11/10/09	Firewalls (Slides)		A quantitative study of firewall configuration errors. A. Wool, IEEE Computer, 37(6):62-67, 2005. (link) Linux iptables HOWTO Rusty Russell. (link) Network Security: Private Communication in a Public World, Chapter 23.
11/12/09	Intrusion Detection		A Sense of Self for UNIX Processes . S. Forrest, S. A. Hofmeyr, A. Somayaji, T. A. Longstaff, In Proceedings of the IEEE Symposium on Security and Privacy, 1996. (link) The Base-Rate Fallacy and its Implications for the Difficulty of Intrusion Detection. S.

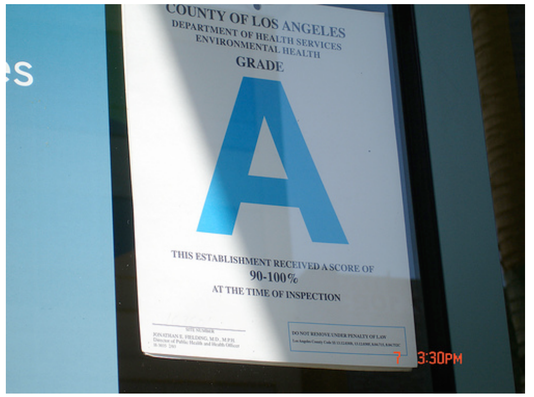
Other Communication Medium

- Slack
  - Will be set up this week and you will be notified there.
  - Find project partners and discussion buddies.

Grading

- The course will be graded on projects, paper reviews, presentations, and class participation in the following proportions:

50% Research Project
25% Paper Presentation
15% Paper reviews
10% Participation



- NOTE: Must do better than 50% average on **each of part** to pass the course.

Projects

- https://syed-rafiul-hussain.github.io/index.php/teaching/cse597-s25/projects.html
- Goal: Learn security principles
- Goal: Learn security analysis techniques
- Goal: Learn research skills

- Projects (Individual or Group)
  - Software Security
    - Attacks
  - Network Security
    - Attacks and Defenses
- Lateness Policy:

Readings

- There are a large amount of readings in this course covering various topics. These assignments are intended to:
  - Support the lectures in the course (provide clarity)
  - Augment the lectures and provide a broader exposure to security topics.
- Students are ***required*** to do the reading!
-

Paper reviews

- Goal: Record key ideas and methods for later
- We will review one paper per weekPurpose:
  - get paper's contributions (what?)
  - understand the techniques (how?)
  - critically analyze the worthiness of the paper
  - where it fits in to the existing body of knowledge

Assignment 0

- This assignment is to help me get to know you and what you expect to achieve in this class. There are no right or wrong answers. Please make sure to limit your answers within 1 page.
  - Briefly mention about yourself and your research interests. What have you studied in the past?
  - Describe your prior experience with computer security if you have any, why you are interested in this class, what you would like to achieve/learn throughout the course, and if you like to pursue a career in security. It's also okay to mention that you are taking the course to fulfill the degree requirements.
  - Describe what you think are the three most recent important issues in computer security and privacy.important issues in computer security and privacy.

Ethics Statement

- This course considers topics involving personal and public privacy and security. As part of this investigation we will cover technologies whose abuse may infringe on the rights of others. As an instructor, I rely on the ethical use of these technologies. Unethical use may include circumvention of existing security or privacy measurements for any purpose, or the dissemination, promotion, or exploitation of vulnerabilities of these services. Exceptions to these guidelines may occur in the process of reporting vulnerabilities through public and authoritative channels. Any activity outside the letter or spirit of these guidelines will be reported to the proper authorities and may result in dismissal from the class and or institution.
- When in doubt, please contact the instructor for advice. Do not undertake any action which could be perceived as technology misuse anywhere and/or under any circumstances unless you have received explicit permission from Professor Hussain.

Academic Integrity Policy

- See the EECS Department's Academic Integrity Standards for CMPSC, CMPEN, and CSE Programming Courses
- http://www.eecs.psu.edu/students/resources/EECS-CSE-Academic-Integrity.aspx
- You must follow this policy

Academic Integrity Policy

- The Department of Computer Science and Engineering expects all student programming work assigned in a class to be completed independently by students (or by teams if permitted/required) and to consist of code designed and developed solely by the students. The use of any other code is not permitted unless the course instructor explicitly allows it and such code is clearly identified as coming from an external source and that source is credited. Students will never be given credit for code which they did not construct.
- The department uses software tools to identify similarities in code submitted by students. These tools differentiate between insignificant cosmetic differences (names used in code, the order of certain code elements) and significant structural similarities (algorithms, data organization). These tools give a percentage of common code between two submissions and identify this common code. We do not set a single, fixed percentage above which we automatically determine that an academic violation has occurred. Rather we rely on the expertise of the instructor to determine when similarities rise above what a reasonable person could expect two students working independently to construct.

Academic Integrity Policy

- For example, in an introductory course in which the programming assignments require relatively short solutions (i.e., less than 50 lines of code) we would expect to see similarities in student solutions rising to a significant percentage of the code. But in an advanced course in which programming projects may contain thousands of lines of code, only a small percentage may be similar but still constitute an academic integrity violation if the code in question was a significant/important aspect of the assignment and if the similarities found could not, in the opinion of the instructor, have been independently developed.
- Furthermore, in cases where student submissions have been found to contain significant portions of code found in online sources (e.g., a common code hosting site is GitHub), the determination of an academic integrity violation is essentially automatic.

Other Course Policies

- Class Recording Policy
  - Video and audio recordings of class lectures will be part of the classroom activity. The video and audio recording is used for educational use/purposes and only may be made available to all students presently enrolled in the class. For purposes where the recordings will be used in future class session/lectures, any type of identifying information will be adequately removed.
- Copyright Policy
  - All course materials students receive or to which students have online access are protected by copyright laws. For courses in which they have previously been or are currently enrolled, students may use course materials and make copies for their own use as needed, but unauthorized distribution and/or uploading of materials without the instructor's express permission is strictly prohibited.

Roadmap

- Formal verification
  - Propositional Logic, First order Logic (FOL), Quantifier-free FOL, Linear temporal Logic
  - SAT/SMT, Model Checker, Runtime Verification, Deductive Verification
- Static Program Analysis
  - Control and data flow analysis
- Dynamic Program Analysis
  - Fuzzing, Symbolic Execution

https://syed-rafiul-hussain.github.io/index.php/teaching/cse597-s21/schedule.html

Adversary

- An adversary is any entity trying to circumvent the security infrastructure
  - The curious and otherwise generally clueless (e.g., script-kiddies)
  - Casual attackers seeking to understand systems
  - Venal people with an axe to grind
  - Malicious groups of largely sophisticated users (e.g, chaos clubs)
  - Competitors (industrial espionage)
  - Governments (seeking to monitor activities)

Thinking Like an Adversary

- Computer security experts think like an attacker all the time
  - "What can go wrong?"
  - "How can it go wrong?"
  - "What assumptions might not be correct?"
  - "How can I exploit the system?"

Thinking Like a Defender

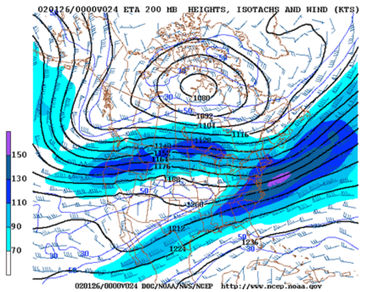
- Security Policy
- Threat Model
- Risk Assessment
- Countermeasures

Trust

- Trust refers to the degree to which a principal is expected to behave
  - What the principal not expected to do?
    - E.g., not expose password
  - What the principal is expected to do (obligations)?
    - E.g., obtain permission, refresh
- A trust model describes, for a particular environment, who is trusted to do what?
- Note: you make trust decisions every day
  - Q: What are they?
  - Q: Whom do you trust?

Security Model

- A security model is the combination of a trust and threat models that address the set of perceived risks
  - The "security requirements" used to develop some cogent and comprehensive design
  - Every design must have security model
    - LAN network or global information system
    - Java applet or operating system
- This class is going to talk a lot about security models
  - What are the security concerns (risks)?
  - Who are our adversaries?
  - What are the threats?
  - Who do we trust and to do what?
- Systems must be explicit to be secure.



The Security Mindset

- Thinking like an attacker
  - Understanding how to circumvent security
  - Look for where security can fall down
- Thinking like a defender
  - What are you defending and from whom
  - Weigh benefits vs. costs: No system is ever completely secure!