CS165 – Computer Security

Introduction

September 26, 2025

Outline

- Welcome!
- Goals of this course
  - Get to know computer security
  - Master a subset of critical skills in security
- Grading
- Introduction to the class

Course Intro

- Trent Jaeger, CSE Prof.
- Email: trentj@ucr.edu
- eLearn used for announcements, assignments, forum
- Webpage: https://www.cs.ucr.edu/~trentj/cs165-f25/ (same as Syllabus in eLearn)
  - Link to the course schedule
- Office: WCH 442
- Office Hours: 4-5pm M and 3-4pm W or by appointment
- TA: Rohith Krishna Papani rohithkrishna.papani@email.ucr.edu
  - Office Hours: 3-4pm Tu and Fr in WCH 368

Self Intro

- Trent Jaeger, CSE Prof.
- Education
  - PhD, University of Michigan
  - BSE, Cal Poly Pomona
- Work
  - AI Development, EDS (GM → HP → DXC → Perspecta)
  - OS and Security Research, IBM Research
  - Professor at Penn State
  - Professor at UC Riverside
- Research in Software and Operating Systems Security

Problem – Compromise the Internet

2000s – One vulnerability can take over the Internet

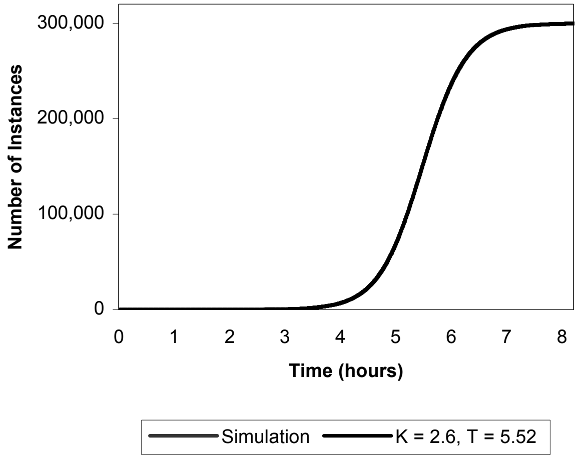
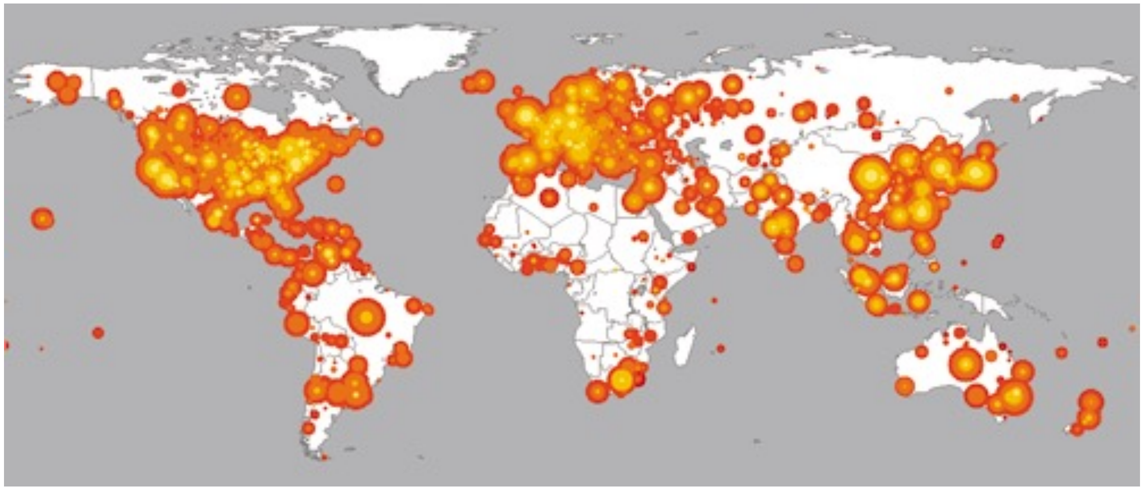


Figure 6: The spread of a simulated worm capable of 10 scans/second in a population of 300,000 vulnerable machines and its comparison to the model developed in Section 2. The simulation and theoretical results overlap completely.

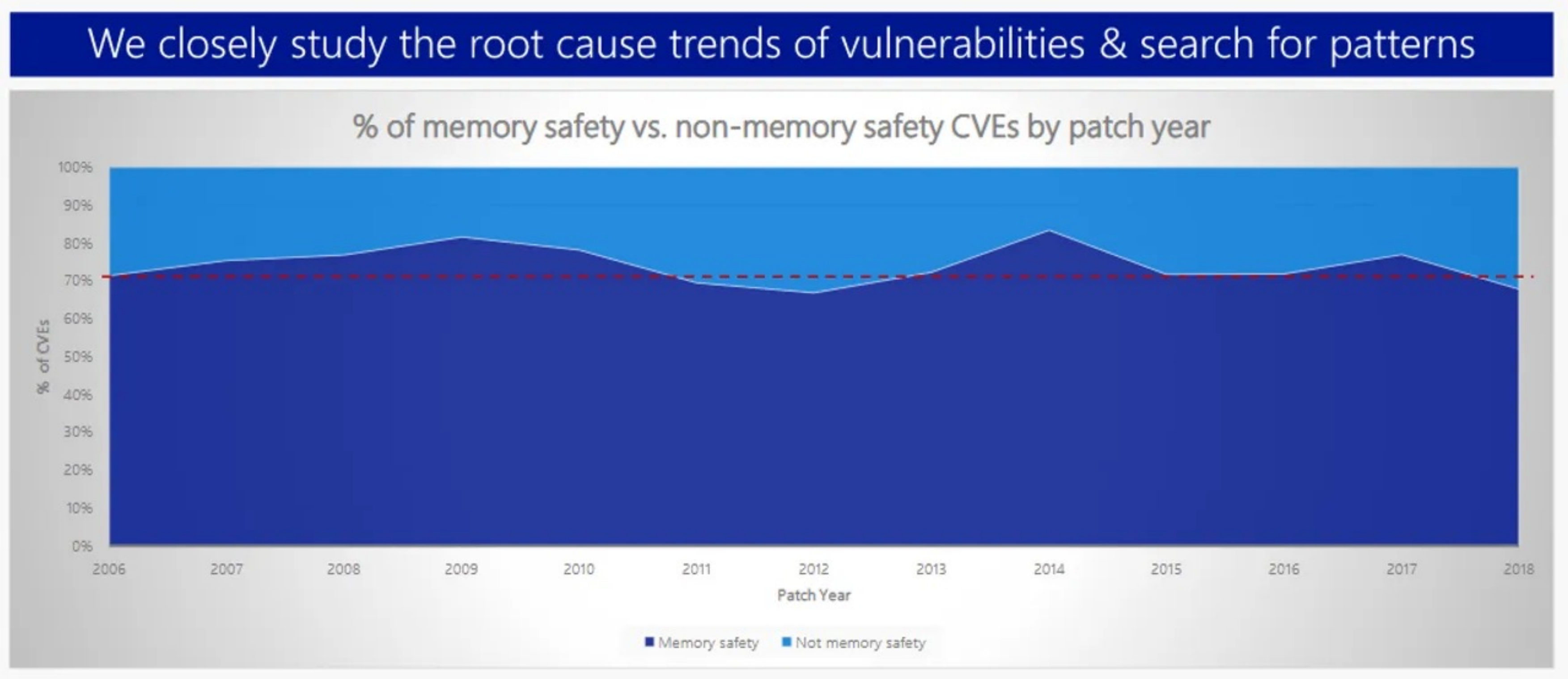


Reaction – Confine Network Programs

2000s – Sandbox "network-facing daemons" to prevent host compromise

Limitation – Buggy Software

Lots of vulnerabilities in software – lots of "memory errors"



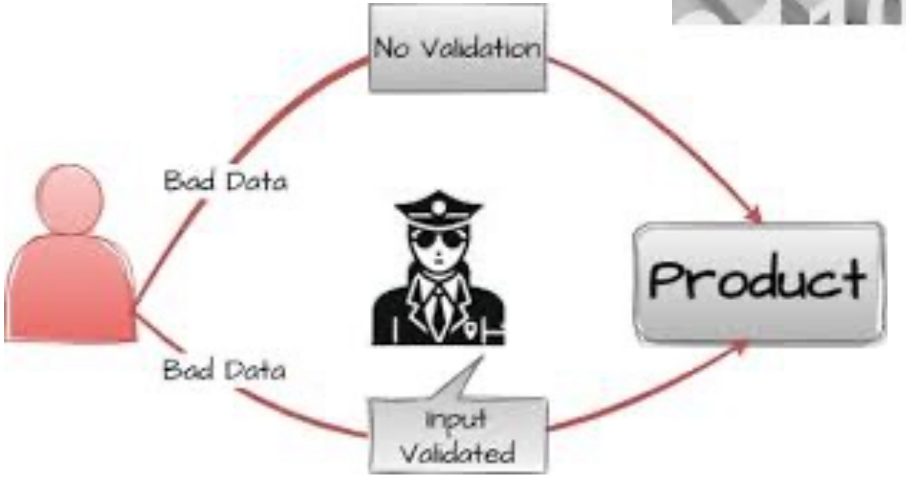
We closely study the root cause trends of vulnerabilities & search for patterns

% of memory safety vs. non-memory safety CVEs by patch year

Exploitation – Mix Code and Data

Use Adversary-Provided Data to Direct Program Execution

www.shutterstock.com · 448121869



Goals of this course

- Learn the principles of computer security
  - Diverse and domain-specific
- Gain hands-on experience (not just theory)
  - Learn to break things
    - $2M bounty for remote jailbreak!
  - And secure systems
    - Write secure code, configure systems securely
    - Analyze a system critically for weaknesses

Goals of this course

- From this class, you’ll learn to:
  - Perform attacks to exploit programs
  - Detect vulnerabilities to prevent exploitation
  - Learn mechanisms to limit impact of exploits
- Become aware of security as a key dimension of software and system design and implementation
- Not an easy course, but you will learn information that will help you throughout your career

Outline

- □ Welcome!
- □ Goals of this course
- □ Grading
- □ Introduction of the class

Course Info and Readings

- Course Management via Elearn (Canvas)
- Textbook
  - Tools and Jewels. Paul van Oorschot. Online.
- Readings for each class determined by the class schedule
  - https://www.cs.ucr.edu/~trentj/cs165-f25/schedule.html
    - Linked from syllabus in elearn and …
    - https://www.cs.ucr.edu/~trentj/cs165-f25/
- Optional Textbook
  - Hacking: The Art of Exploitation (2nd Edition), by Jon Erickson
    - *Useful for understanding attacks.*

Grading

- 1 Midterm: 25%
- 1 Final: 30%
- 3 Projects: 25%
- 4 Quizzes (in class): 15% (drop lowest)
- Participation: 5%
  - Attendance and participation in class, project participation, intellectual contribution
- Course will proceed quickly, so it is important to keep up with assignments/readings

More Detail

- Quizzes
  - In class: 1-2 questions, based on lectures/homeworks
  - Drop lowest
  - Can give necessary time, but accommodations available
- Projects
  - Programming, Exploitation, Testing & Securing
  - In teams of 2
  - Will assume you both have completed the entire project
- Exams
  - Test material broadly/deeply – average in low 70s usually
  - Need to prepare via quizzes, homeworks, projects

Late policy

- 4 slack days for homework or project (combined)
- 2% bonus points if you do not use any
  - All or nothing
- When you exhaust your slack days, there is a 20% per day per assignment score deduction
  - If: (1) all the slack days have been exhausted and (2) that assignment is late.

Academic Integrity

- UCR Academic Integrity Polices & Procedures
  - Linked to syllabus
  - Do not share your work with others in the class!

Ethics

- This course considers topics involving software exploitation techniques. 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.

- When in doubt, please contact the instructor for advice. Do not undertake any action that could be perceived as technology misuse anywhere and/or under any circumstances unless you have received explicit permission from Professor Jaeger.

Laws and Policies

- University of California Electronic Communications Policy
  - You can be expelled
- Federal and state laws criminalize computer intrusion and wiretapping
  - e.g., Computer Fraud and Abuse Act (CFAA), Electronic Communications Privacy Act (ECPA)

Getting an A

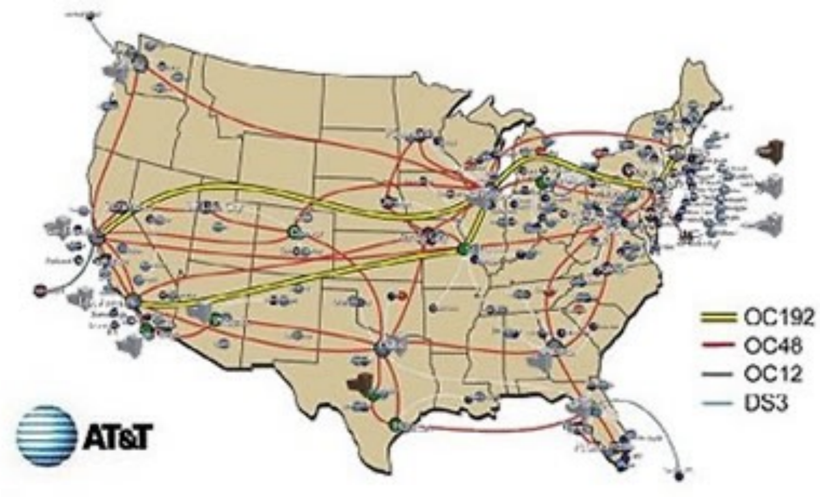
- Background
  - This class requires knowledge of computer organization, operating systems, networking
  - CS61, CS161, CS153, CS164 (a little bit)
  - And a mature understanding of software and systems in general
- During the course
  - Keep up on readings and assignments
  - Projects and quizzes aim to prepare you for exams
  - Exams are the most important part, and will be very difficult if you are not prepared

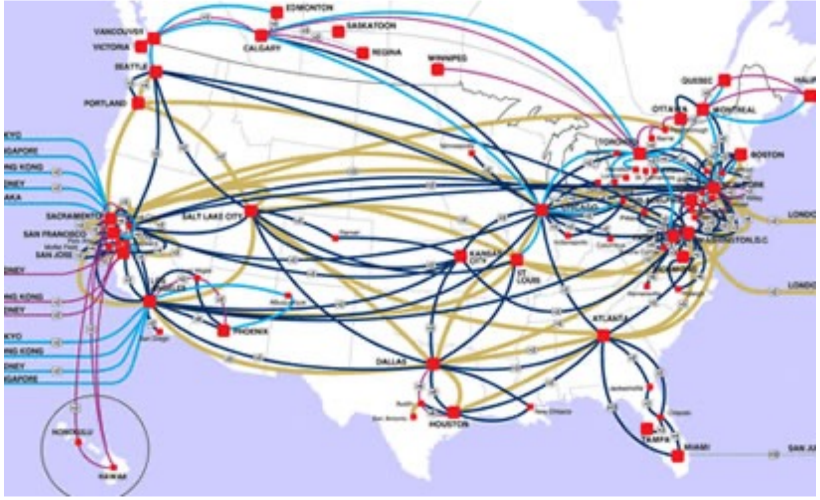
Outline

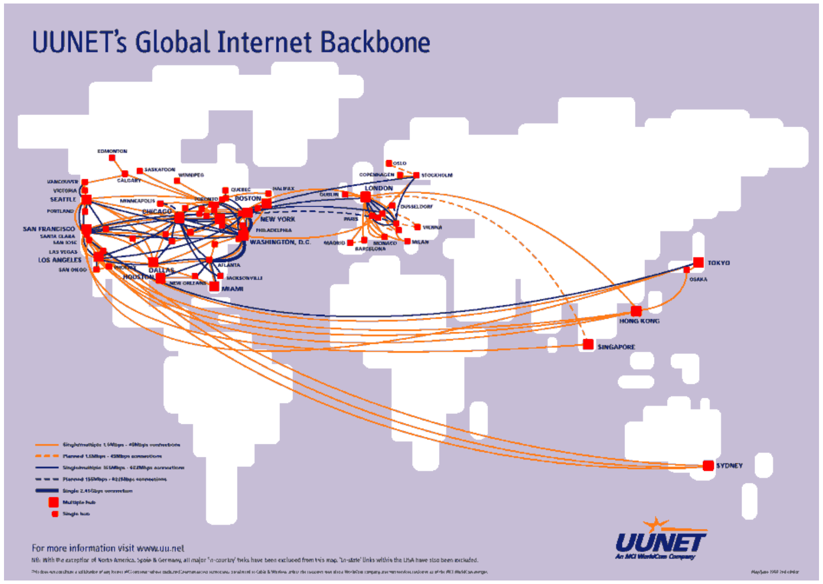
- □ Welcome!
- □ Goals of this course
- □ Grading
- □ Introduction of the class

Computer

Network


OC192
OC48
OC12
DS3
AT&T


EDMONTON
VANCOUVER
VICTORIA
CALGARY
SASKATOON
REGINA
WINNIPEG
SEATTLE
PORTLAND
QUEBEC
OTTAWA
MONTREAL
BOSTON
SACRAMENTO
SAN FRANCISCO
SAN JOSE
SALT LAKE CITY
PHOENIX
DALLAS
HOUSTON
ATLANTA
WASHINGTON D.C.
TAMPA
MIAMI
HONOLULU
HAWAII


UUNET's Global Internet Backbone
EDMONTON
SASKATOON
CALGARY
VANCOUVER
VICTORIA
SEATTLE
PORTLAND
SAN FRANCISCO
SANTA CLARA
SAN JOSE
LAS VEGAS
LOS ANGELES
SAN DIEGO
WINNIPEG
MINNEAPOLIS
CHICAGO
TORONTO
QUEBEC
HALIFAX
BOSTON
NEW YORK
PHILADELPHIA
WASHINGTON, D.C.
DALLAS
HOUSTON
ATLANTA
NEW ORLEANS
JACKSONVILLE
MIAMI
OSLO
STOCKHOLM
COPENHAGEN
DUBLIN
LONDON
DUSSELDORF
PARIS
VIENNA
MADRID
MONACO
BARCELONA
MILAN
TOKYO
OSAKA
HONG KONG
SINGAPORE
SYDNEY
Single/multiple 1.5Mbps - 45Mbps connections
Planned 1.5Mbps - 45Mbps connections
Single/multiple 155Mbps - 622Mbps connections
Planned 155Mbps - 622Mbps connections
Single 2.45Gbps connection
Multiple hub
Single hub
For more information visit www.uu.net
NB: With the exception of North America, Spain & Germany, all major 'in-country' links have been excluded from this map. 'In-state' links within the USA have also been excluded.
UUNET
An MCI WorldCom Company

Physical Sensing

Actuation Information

Object Domain

/is/labs/lim-lab/image/4.CPS.jpg

SECURITY RISK
HOW SAFE IS YOUR COMPUTER?

IS IT SAFE?

Recent news

Data breach in March 2023

Facebook 50M accounts

Dyn DDoS attack

Insider threats

What is security about?

Security and security mindset

- "The study of how a system behaves under adversarial actions"
  - Intelligent attackers actively trying to lead the system to misbehave or do unexpected things

- Security vs. System

  || ||

- Corner cases vs. Common case

Play Games vs. Security Games

Not enough psi...build more Pylons.

Play Games vs. Security Games

- Both deal with a set of man-made rules!
- Man-made rules have bugs (which can be exploited)!
  - Think about tax systems…
  - Warren Buffett’s tax rate is lower than his secretary’s
    - 17.3% on $39.8 million taxable income
    - Heck, it’s lower than my tax rate
  - The more complex, the more dangerous

Thinking like an attacker

- Analyze game rules with goals to circumvent
  - Break into a locked door? Fake an identity?
  - Thought experiment: How to get into offices after hours?
- Think outside the box
  - Make program run differently than expected
  - Exploit unexpected effects
- Challenge security assumptions
  - All available points of attack: Threat Model
- One successful attack that exploits a vulnerability is good enough!

Thinking like a defender

- Discover loopholes in game rules and fix them
  - With respect to a Threat Model
  - Need to cover all corner cases (is currently HARD!)
  - Always catch-up
- Design favorable rules for defense
  - Prevention of bad consequences (also HARD!)
  - Need to allow legitimate functionalities (which may lead to bad consequences)

Topics

- Software security
  - Vulnerabilities
  - Causes
  - Exploits
  - Defenses
- Systems security
  - File security
  - Web security
  - Network security
- Future!

Not a theory
or
crypto class

Questions