EECS 388
Introduction to Computer Security

Welcome; The Security Mindset
Jan 7, 2015

Today's Class

- Welcome!
- Goals for the course
- Security Mindset
  - Thinking like an attacker
  - Thinking as a defender
- Course mechanics
- Ethics

Who are we?

Z. Morley Mao

CSE Professor

Web: eecs.umich.edu/~zmao

Email: zmao@umich

Office: 4629 Beyster

Eric Wustrow

CSE Ph.D. candidate

Web: ericw.us/trow

Email: ewust@umich

Office: 4828 Beyster

Hours: Mon 1-2PM, Wed 4-5PM

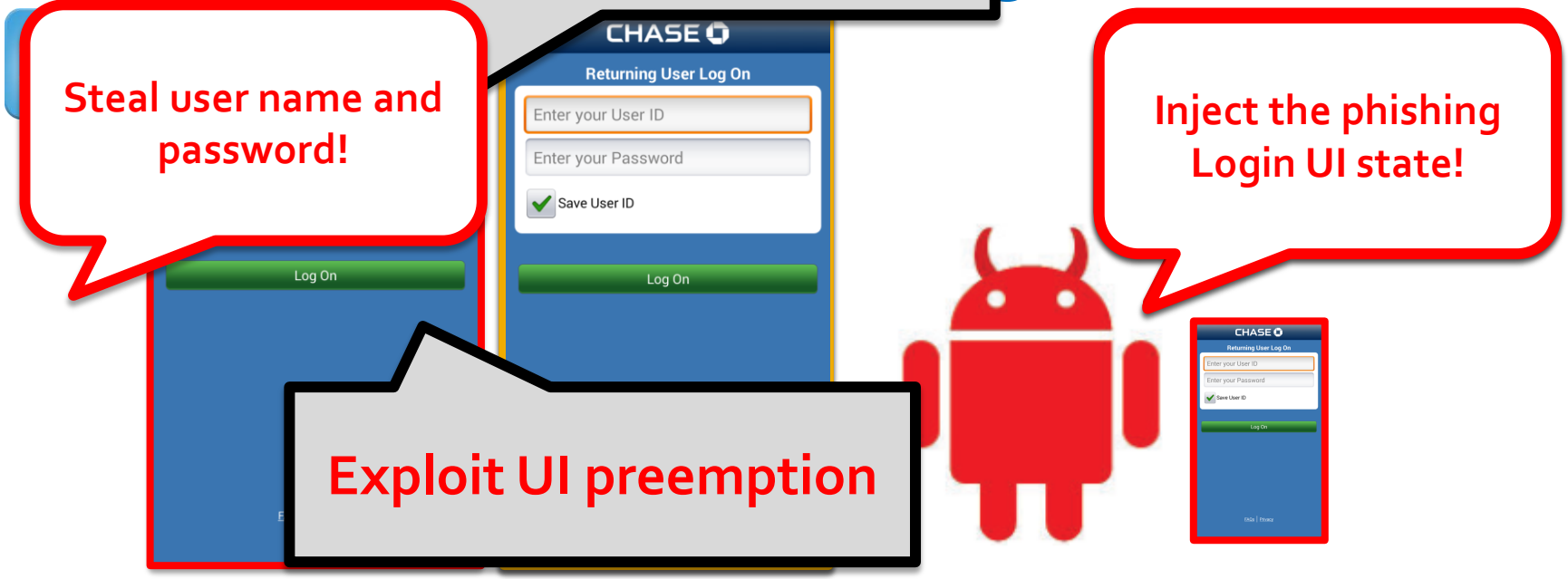
Who are we?

- Z. Morley Mao
- CSE Prof. in software lab
- UC Berkeley Ph.D.
- http://www.eecs.umich.edu/~zmao
- Office: 4629 Beyster
- Research interests: networks, mobile computing, network/system security.

Android security: UI State Hijacking

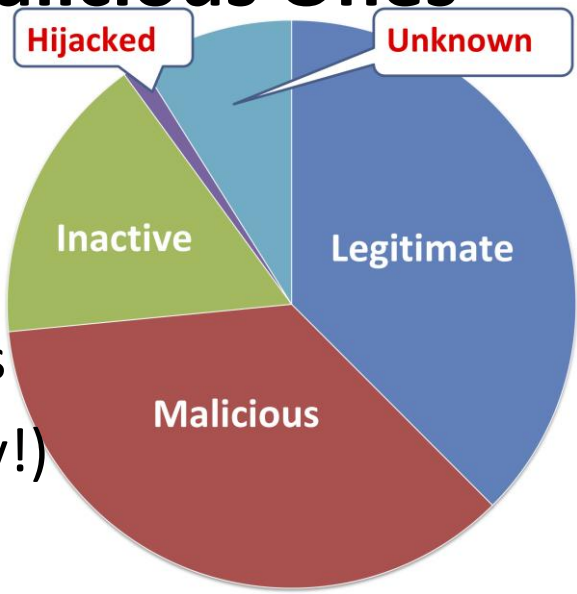
- Hijack ser… l private input

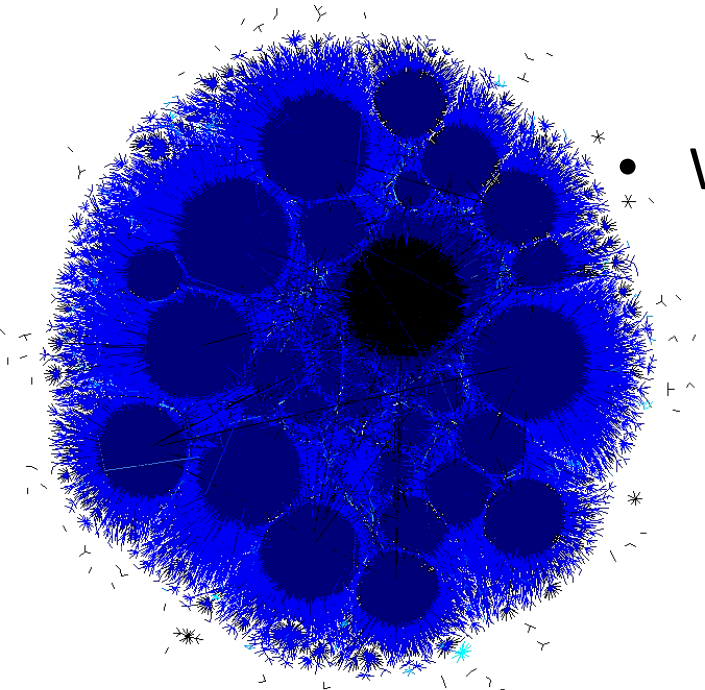
Foreg… ground:



SocialWatch: Legitimate Users of Online Services are Socially-Connected, not for Malicious Ones

- Combating attackers are **challenging** for online services (e.g. email):
  * **682 million** users and **5.745 billion** edges.
  * Attackers **evolve** developing counter strategies
  * Existence of **hijacked** users (hard-to-detect spy!)

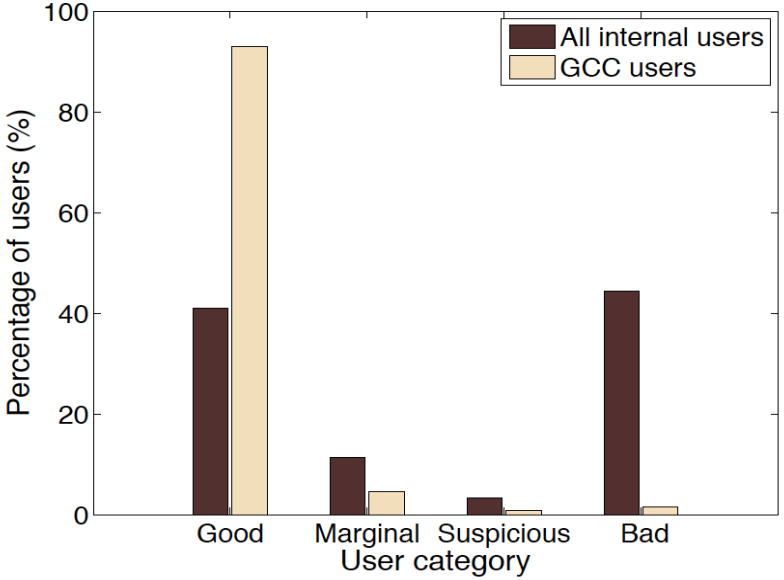




Visualization of 295K sampled good users

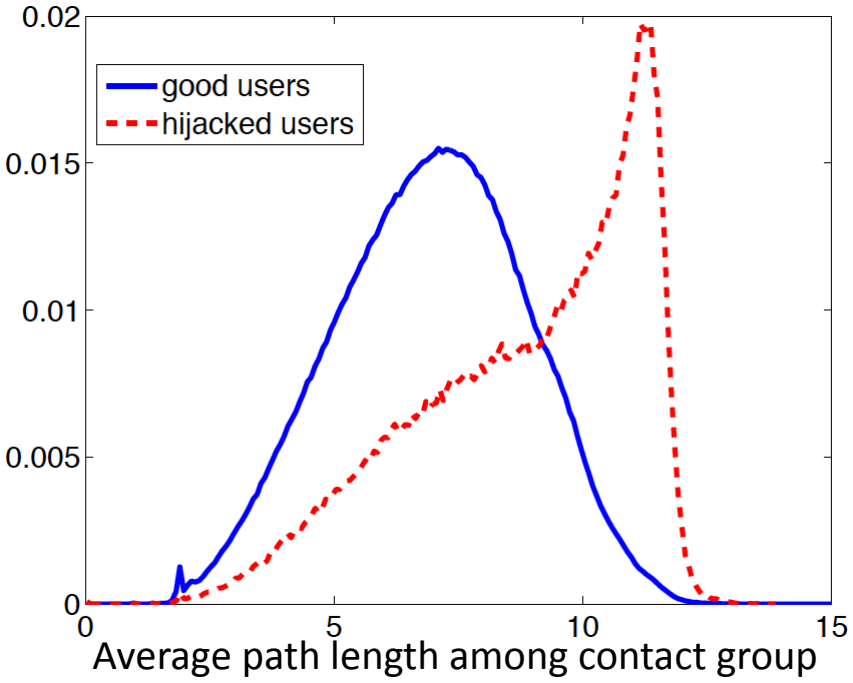
- What's fundamentally unique for legitimate?
  * **Socially-connected** (small-world theory) forming a big "**island**"
  * People they talk to may **inter-connect**
  * In the social graph, their contacts are **close** in distance

SocialWatch: Online Service Protection System with Social Graphs



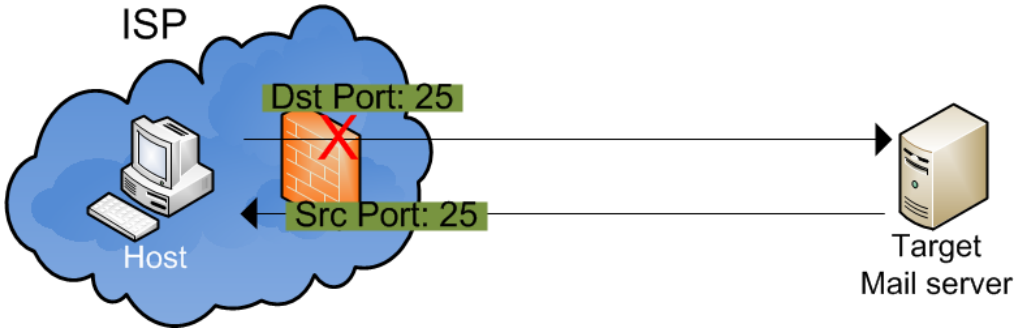
- **GCC** (Giant Connected Component) users are mostly legitimate
  * Good-user profile computation
  * Good users talk to good users (legitimate user expansion)

- Robust **social features** to detect **malicious / hijacked** users
- Classify **101.1 million** legitimate users, detect **1.1 million** hijacked users & **11.9 million** malicious users (previous unknown) with **< 1%** false positive



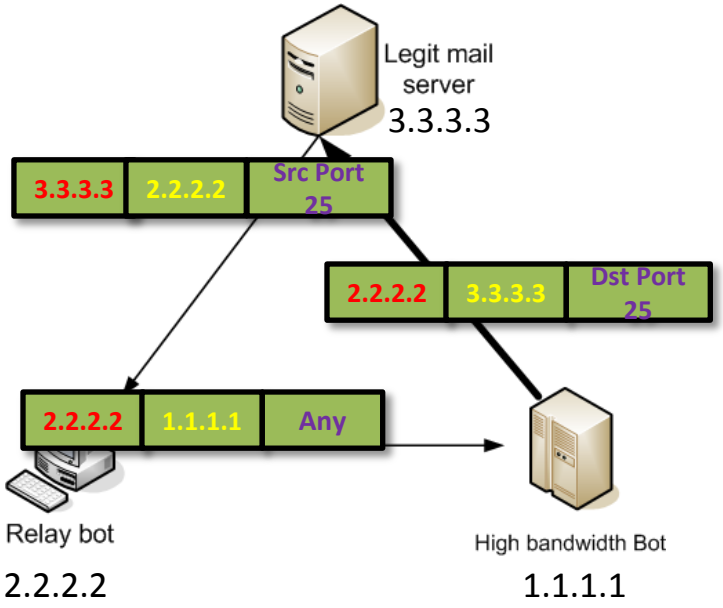
Discovered a New Stealthy Spamming Technique

- Hypothesis -- ISPs only block outgoing dest port 25, not incoming source port 25



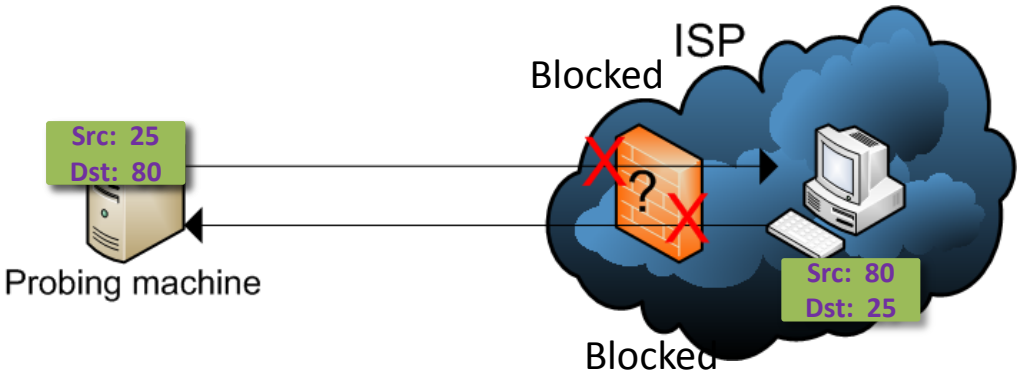
- Can allow their IPs still be used in spamming (relay bot)

Legend



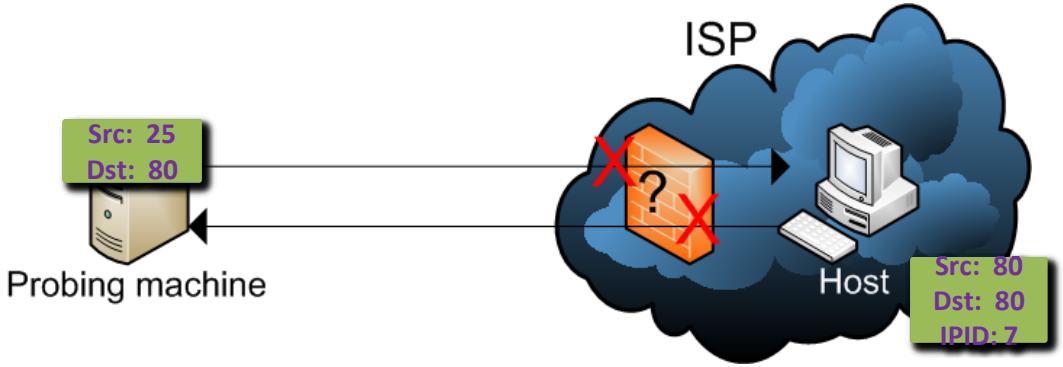
IP spoofing

Detecting One Directional Port 25 Blocking



One-ended probing difficult to know which direction blocking happens

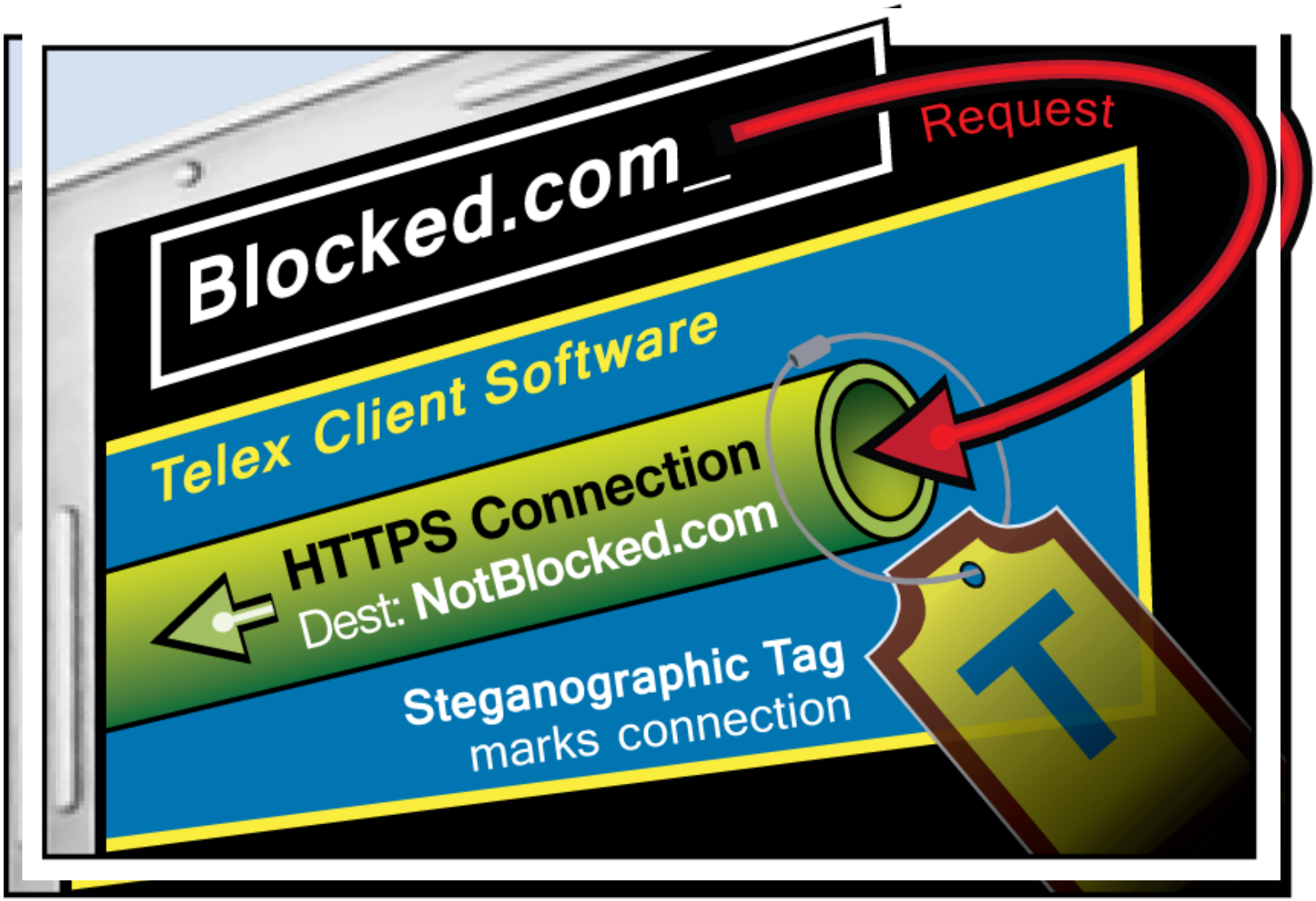
- But, we can use IPID value (identifier in IP header) -- monotonically increasing



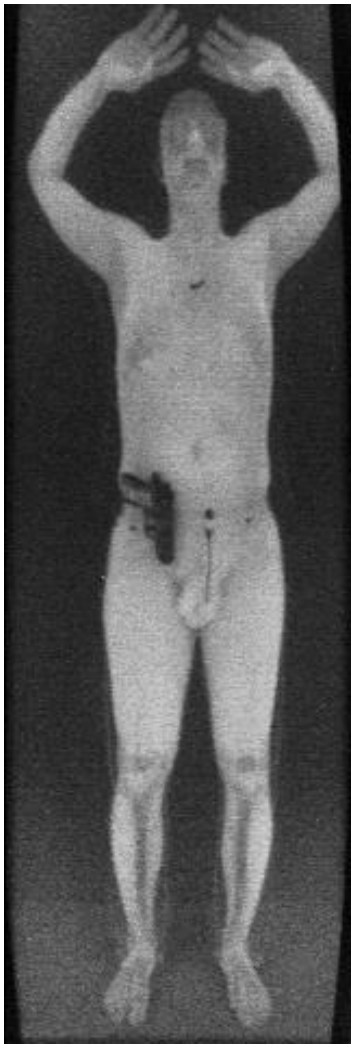
Get the current IPID
Test packets received or not
Infer previous packets received or not

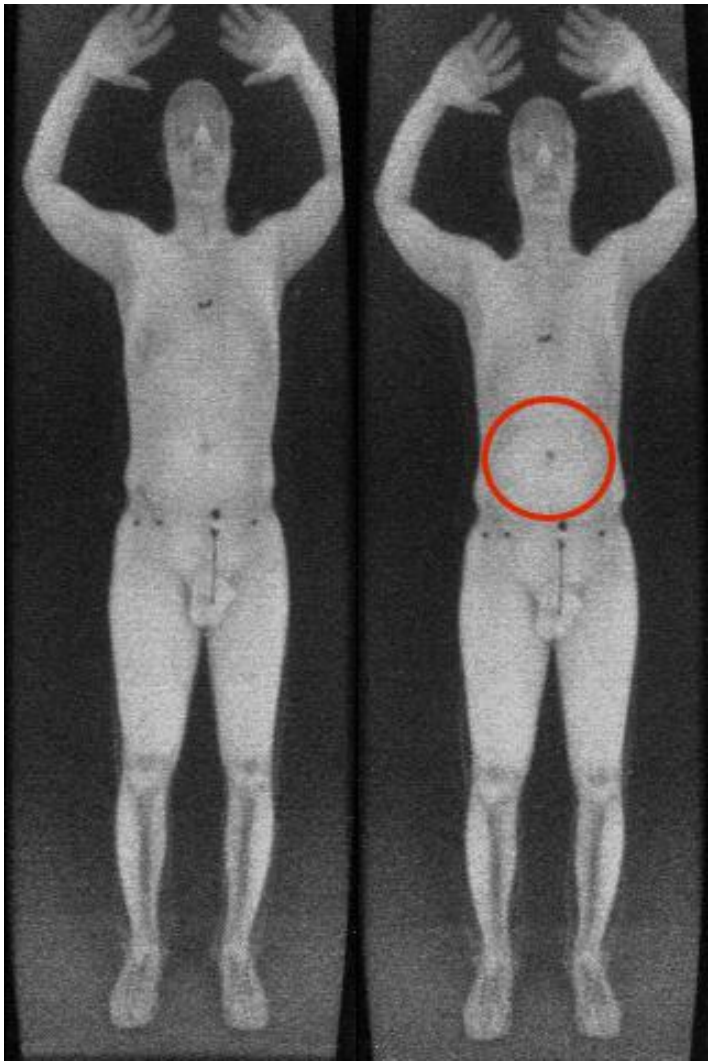
- Results: 97% of 688 studied prefixes indeed only blocked outgoing dest port 25

My work: anticensorship



My work - cyberphysical security

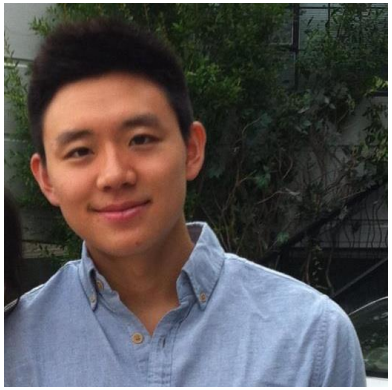




Who are we?

Alishah Chator
alishahc@

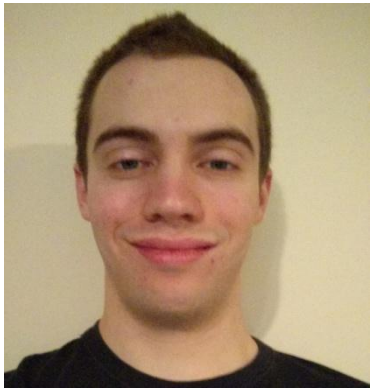
Dylan Hurd
drhurd@



Andrew Lee
ajyl@

Luis Molina
luismol@

Zane Salem
zsalem@



Brad Warren
bradmw@

Peter Xie
xiep@

Course Topics

The Security Mindset	Principles, threat modeling…
Applied Cryptography	Public and private-key cryptography, digital signatures and authentication, hash functions, secure channels…
Internet Security	IP, TCP, routing, network protocols, web architecture, web attacks, Firewalls, intrusion detection
Application Security	Defensive programming, memory protection, sandboxing, virtual machines, buffer overflows, malware
Culture, Law, and Politics	Privacy, security and the law, digital rights management, voting, ethics…

Introduction to Computer Security

Winter 2015

This course teaches the security mindset and introduces the principles and practices of computer security as applied to software, host systems, and networks. It covers the foundations of building, using, and managing secure systems. Topics include standard cryptographic functions and protocols, threats and defenses for real-world systems, incident response, and computer forensics. See the schedule for details.

Professor	Z. Morley Mao, Eric Wustrow Office hours: Mon 1:00–2:00, Wed 4:00–5:00, 4629 Beyster, or by appointment
Prerequisites	EECS 281; EECS 370 recommended
Lectures	Mon./Wed. 9:00–10:30, 10:30–noon, 1670 Beyster
Discussion	Section 11: Fri. 1:30–2:30, 1620 Beyst Section 12: Mon. 1:30–2:30, 1620 Beyst Section 13: Wed. 3:30–4:30, 1620 Beyst Section 14: Tue. 4:00–5:00, 1620 Beyst Section 15: Thu. 3:00–4:00, 1620 Beyst Section 16: Tue. 3:00–4:00, 1620 Beyst
TAs	Andrew Lee, GSI (Office hours: Mon 2:30-3:30PM in BBB Learning Center) Luis Molina, GSI (Office hours: Thu 4-5PM in BBB Learning Center) Peter Xie, GSI (Office hours: Wed 4:30-5:30PM in BBB Learning Center) Alishah Chator, IA (Office hours: Tue 1-2PM in BBB Learning Center) Dylan Hurd, IA (Office hours: Tue 2-3PM in BBB Learning Center) Zane Salem, IA (Office hours: Fri 3:30-4:30, Wed 2:30-3:30PM in BBB Learning Center) Brad Warren, IA (Office hours: Fri 2:30-3:30PM in BBB Learning Center)
Communication	We'll use Piazza for general discussion and questions about course material.

Course Schedule

Winter 2015

This schedule is subject to change. Please check back frequently.

Part 1. Security Fundamentals

Monday Lecture	Wednesday Lecture	Discussion
	Jan. 7 **The security mindset** Threat models, vulnerabilities, attacks; how to think like an attacker and a defender	Introduce Homework 1 Python tutorial
Jan. 12 **Message integrity, pseudorandom functions** Alice and Bob, crypto games, Kerckhoffs's principle, hashes and MACs	Jan. 14 **Randomness and pseudorandomness** Generating randomness, PRGs, one-time pads	Introduce Crypto Project Python tutorial
Jan. 19 Martin Luther King, Jr. Day– No lecture	Jan. 21 **Block ciphers** Simple ciphers, AES, block cipher modes Homework 1 due 6pm	Review Homework 1 Introduce Homework 2
Jan. 26 **Key exchange and key management** Diffie-Hellman key exchange, man-in-the-middle attacks	Jan. 28 **Public-key crypto** RSA encryption, digital signatures, secret sharing Crypto Project due 6pm	Review Crypto Project Introduce Web Project

Goals for this Course

- Critical thinking
  - How to think like an attacker
  - How to reason about threats and risks
  - How to balance security costs and benefits
- Technical skills
  - How to protect yourself
  - How to manage and defend systems
  - How to design and program secure systems
- Learn to be a security-conscious citizen
- Learn to be a `1337 hax0r`, but an ethical one!

Getting to Know You

1. Meet two new people and learn their names.
2. Take your picture and email us:

To: **eecs388-staff@umich.edu**

Subject: ***<your_uniqname>***

> What name should we call you?

> What's your year and major?

> Can you program?
> In C? In Python? In x86 asm?

> What would you like to learn in 388?

5 minutes. **Go!**

What is Computer *Security*?

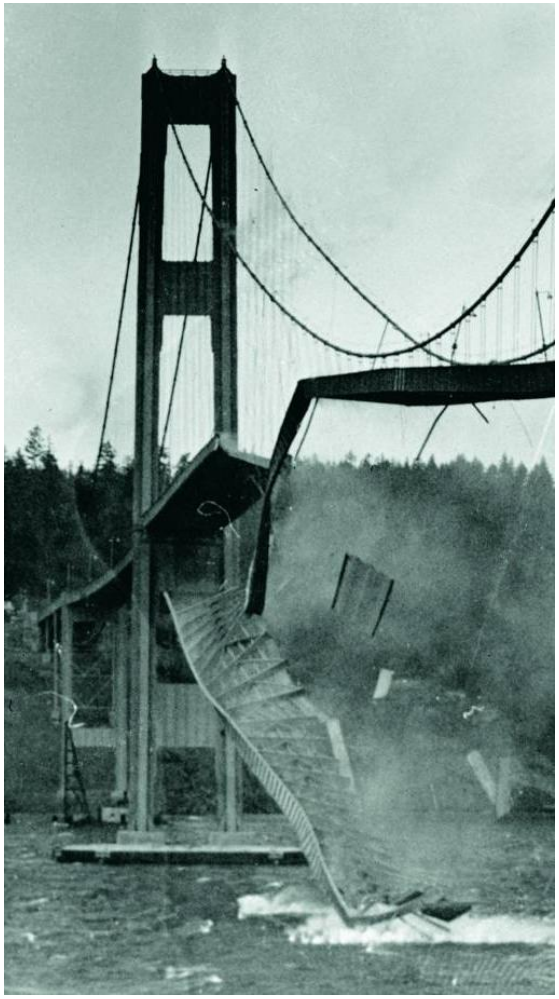
Math?

Engineering?

Philosophy?

Natural Sciences?

What's the Difference?



Meet the Adversary

"Computer security studies how systems behave in the presence of an *adversary*."

The adversary
a.k.a. the attacker
a.k.a. the bad guy

* An *intelligence* that actively tries to cause the system to misbehave.

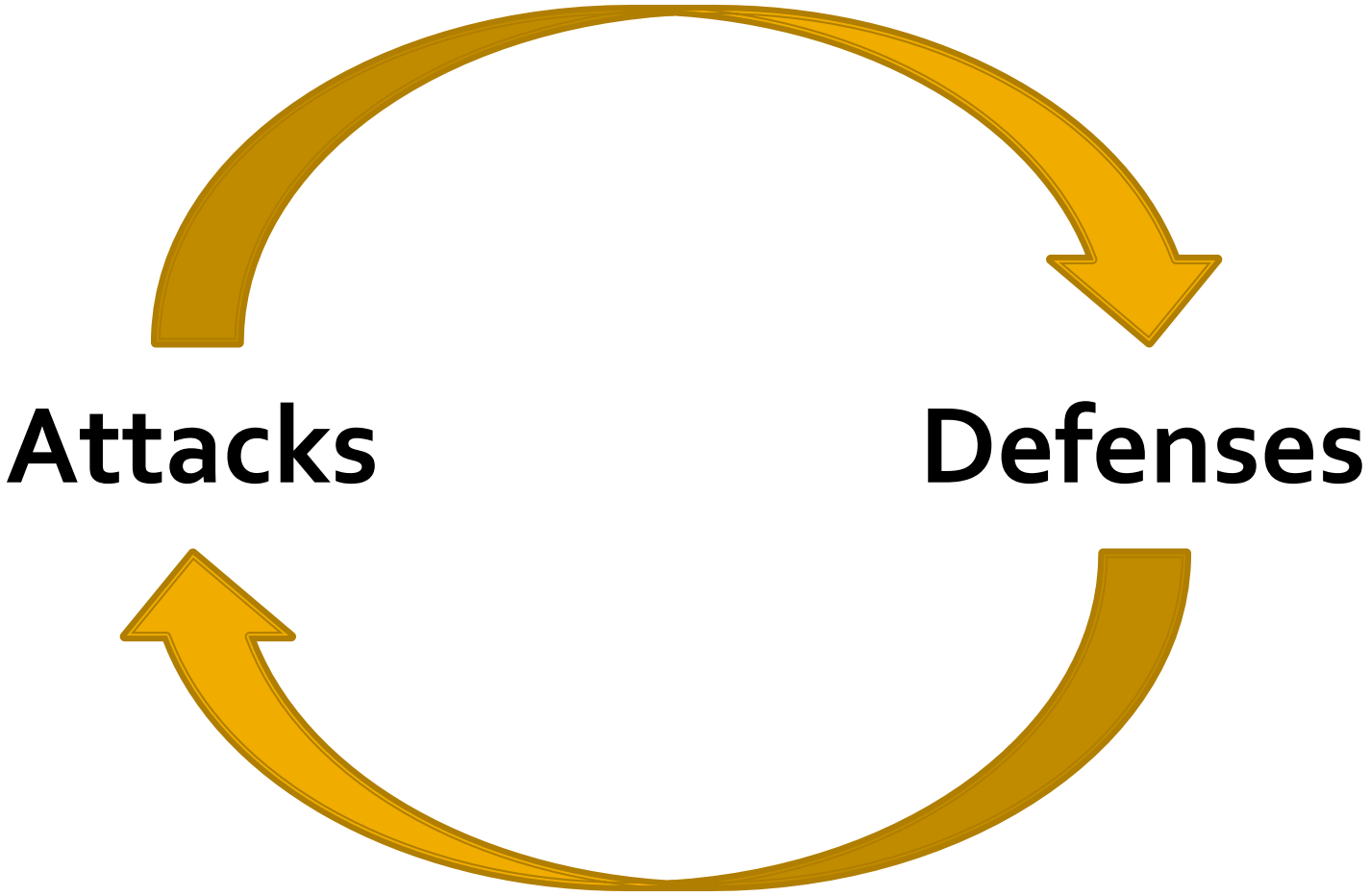
"Know your enemy."

- Motives?
- Capabilities?
- Degrees of access?

The Security Mindset

- Thinking like an attacker
  - Understand techniques for circumventing security.
  - Look for ways security can break,
    not reasons why it won't.
- Thinking like a defender
  - Know what you're defending, and against whom.
  - Weigh benefits vs. costs:
    No system is ever completely secure.
  - "Rational paranoia!"

High-Level Approaches



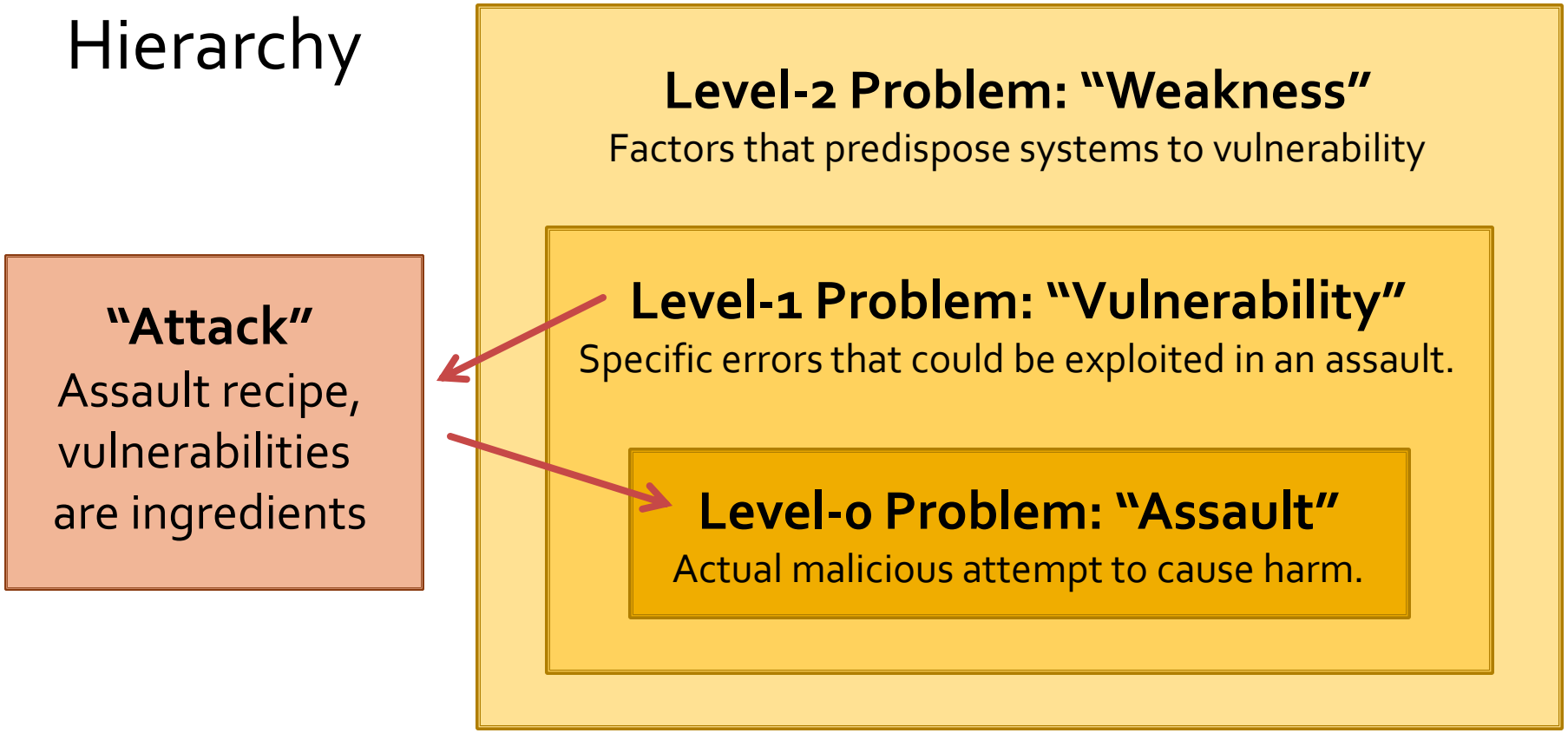
Why Study Attacks?

Identify vulnerabilities so they can be fixed.
Create incentives for vendors to be careful.
Learn about new classes of threats.

- Determine what we need to defend against.
- Help designers build stronger systems.
- Help users more accurately evaluate risk.

"Insecurity"?

Hierarchy



Thinking Like an Attacker

- Look for weakest links – easiest to attack.
- Identify assumptions that security depends on. Are they false?
- Think outside the box: Not constrained by system designer's worldview.

Practice thinking like an attacker: *For every system you interact with, think about what it means for it to be secure, and image how it could be exploited by an attacker*.

Exercises

- Breaking into the CSE building?

Exercises

- What are some security systems you interact with in everyday life?

Thinking as a Defender

- Security policy
  - What are we trying to protect?
  - What properties are we trying to enforce?
- Threat model
  - Who are the attackers? Capabilities? Motivations?
  - What kind of attack are we trying to prevent?
- Risk assessment
  - What are the weaknesses of the system?
  - What will successful attacks cost us?
  - How likely?
- Countermeasures
  - Costs vs. benefits?
  - Technical vs. nontechnical?

Challenge is to think rationally and rigorously about risk. *Rational paranoia.*

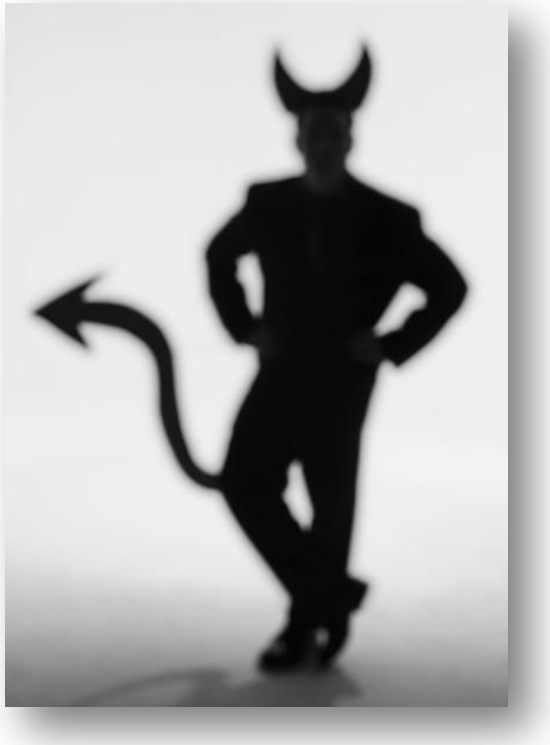
Security Policies

- What *assets* are we trying to protect?
- What properties are we trying to enforce?
  - Confidentiality
  - Integrity
  - Availability
  - Privacy
  - Authenticity
  - ⋮

Threat Models

- Who are our adversaries?
  - Motives?
  - Capabilities?
- What kinds of attacks do we need to prevent?

  (Think like the attacker!)



- Limits: Kinds of attacks we should ignore?

Assessing Risk

Remember: *Controlled* paranoia

- What would security breaches cost us?
  - Direct costs: Money, property, safety, …
  - Indirect costs: Reputation, future business, well being, …
- How likely are these costs?
  - Probability of attacks?
  - Probability of success?

Countermeasures

- Technical countermeasures
- Nontechnical countermeasures

  Law, policy (government, institutional), procedures, training, auditing, incentives, etc.

Security Costs

- No security mechanism is free
  - Direct costs:
    Design, implementation, enforcement,
    false positives
  - Indirect costs:
    Lost productivity, added complexity
- Challenge is rationally weigh costs vs. risk
  - Human psychology makes reasoning about high cost/low probability events hard

Exercises

- Should you lock your door?
  - Assets?
  - Adversaries?
  - Risk assessment?
  - Countermeasures?
  - Costs/benefits?

Exercises

- Should you lock your bike?

Exercises

- Using a credit card safely?

Secure Design

- Common mistake:
  Trying to convince yourself that the system is secure
- Better approach:
  Identify the *weaknesses* of your design and focus on correcting them

- Secure design is a ***process***
  Must be practiced continuously; can't be retrofitted

Where to Focus Defenses

- *Trusted components*
  Parts that must function correctly for the system to be secure.

- *Attack surface*
  Parts of the system exposed to the attacker

- Complexity vs. security?

Other Principles

Defense-in-Depth

Diversity

Maintainability

…

Exercises

- Preventing cheating on the final?

Exercises

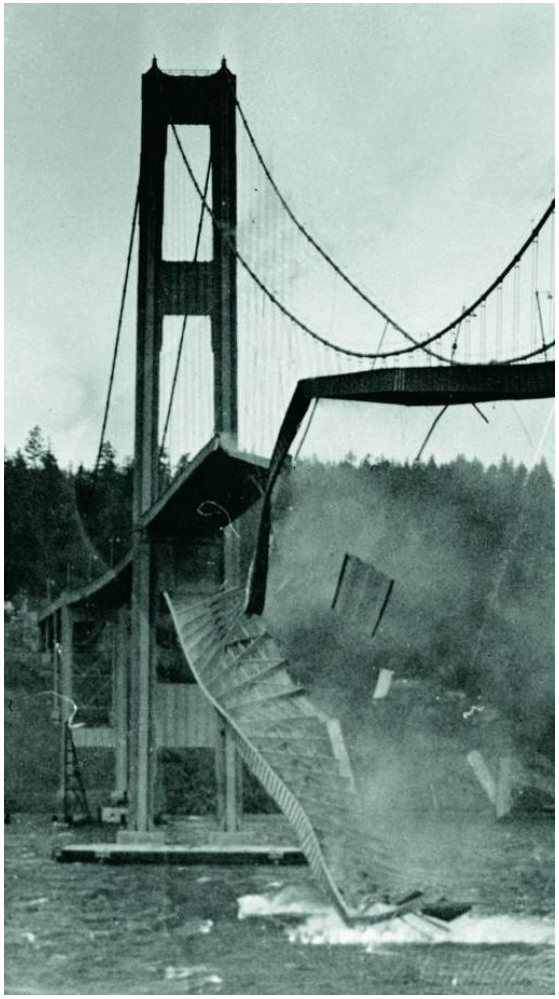
- Preventing you from stealing my password?

Security Testing

- Testing against requirements
  - What are the right requirements?
- Adversarial testing
  - Black box testing
  - White box testing
- Example: Voting

Learning from Failures

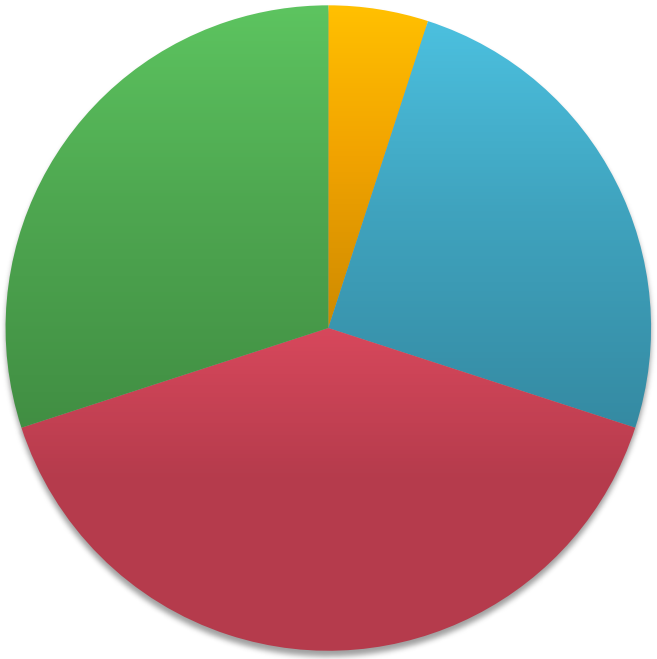
- Time-honored engineering practice, emerging in security
- Identifying *causes* of failures
- What can failure teach us?



Recall Goals for this Course

- Critical thinking
  - How to think like an attacker
  - How to reason about threats and risks
  - How to balance security costs and benefits
- Technical skills
  - How to protect yourself
  - How to manage and defend systems
  - How to design and program secure systems
- Learn to be a security-conscious citizen
- Learn to be a `1337 hax0r`, but an ethical one!

Grading



- Class Participation (5%)
- Homework Exercises (25%)
- Programming Projects (40%)
- Final (30%)

Class Participation (5%)

Attendance
Alertness
Asking questions
Taking part in discussions
Making intellectual contributions

Get points for speaking up and contributing substantial ideas.
Lose for being completely silent, frequently missing class, browsing the web, etc.

Class Participation (5%)

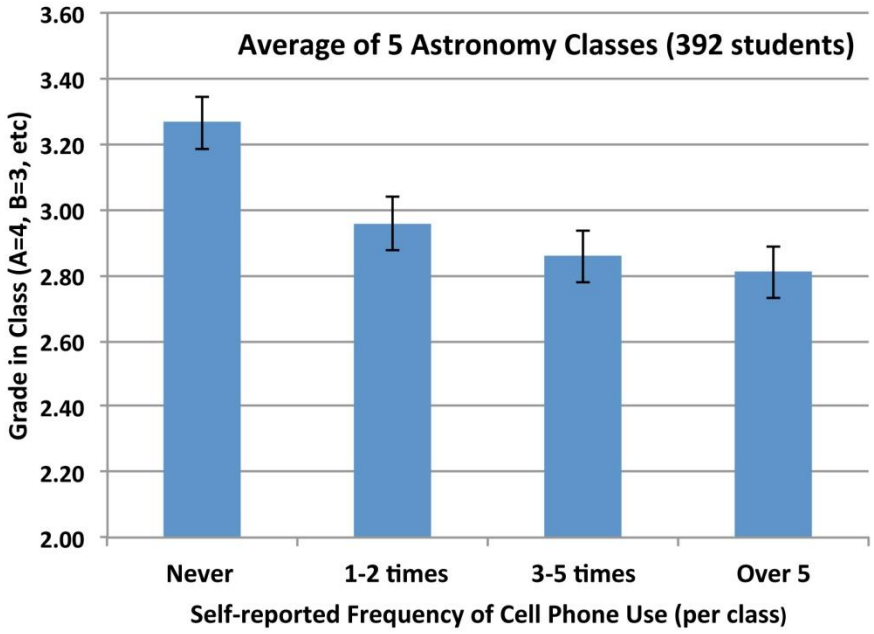
Table 2
Degree to which students felt aspects of the class interfered with their ability to learn lecture material

Item	Mean	SD	*n*
Other students' computer use[a]	3.65	2.35	120
Own computer use[ab]	3.55	2.13	78
Other students talking[bc]	3.16	1.95	119
Length of class[c]	2.98	1.96	120
Other students coming, going, and fidgeting[c]	2.75	1.60	121
Style of class (primarily lecture)[d]	2.26	1.96	119
Time of day[e]	1.96	1.57	120
Classroom environment[e]	1.88	1.35	120
Instructor's use of PowerPoint[f]	1.37	.79	120

Higher numbers indicate greater reported levels of interference.
Note: Items that share a superscript do not differ at the .02 level, based on pairwise comparisons.

Class Participation (5%)

Is there a correlation between use of cell phones during lecture and student grades?



- 75% of students surveyed use a phone during lecture.
- Their frequency of use averaged 7x per hour
- Their class grades averaged 0.36 ± 0.08 **lower**. (A=4.0, B=3.0… scale)

Research by D. Duncan, A. Hoekstra, B. Wilcox, Univ. of Colorado, 2012

Homework Exercises (25%)

Five sets of exercises, done individually.
Problems or short written analysis.
Will be posted on course site.
Can discuss problems approaches,
but complete and turn in own work.
Submit everything via CTools.

- Homework 1 available today.
  Due Wednesday, Jan 21 at 6 p.m.

Programming Projects (40%)

Five programming projects, working in pairs:

1. Cryptography
2. Web security
3. Network security
4. Application security
5. Computer forensics

Will be posted on course site.
Start early! Go to discussions!

- Crypto Project available next Monday.
  Due Wednesday, Jan 28 at 6 p.m.

Lateness Policy

Our constraints:

Return graded work promptly.

Go over solutions in the following week's discussion.

Strict lateness policy:

- 10% penalty for being late.
- Lose additional 10% every 5 hours.
- One penalty-free late exception
- Can't accept work after 20 hours.
- Extensions in *extraordinary* circumstances only.

Please start early!

Collaboration Policy

- Encourage you to help each other learn. You may give or receive help on *concepts.* However, all written work/code must be done by you/your team.
- Cheating is when you give/receive an unfair advantage on assigned work.
- Questions? See us.

- **No cheating!**
  (If we catch you cheating, we won't tell you until after the exam.)

Final Exam (30%)

During exam period.

Covers the entire course.

Similar format to homework questions.

Lectures

Come to lecture.

Take notes! Works best if hand written.

Interrupt, ask questions, share stories.

Lab/Discussion Sections

- Learn secure programming techniques, vital information for completing projects, detailed reviews of completed assignments.

- *Discussions start today!*

Communication

EECS388.org ..... *overview, schedule, assignments*

Piazza ..... *questions, discussion, announcements*

Ctools ..... *homework/project submission*

eecs388-staff@umich.edu .... *administrative issues*

Law and Ethics

- **Don't be evil!**
  - Ethics requires you to refrain from doing harm
  - Always respect privacy and property rights
  - Otherwise you will fail the course
- Federal and state laws criminalize computer intrusion and wiretapping
  - e.g. Computer Fraud and Abuse Act (CFAA)
  - You can be sued or go to jail
- University policies prohibit tampering with campus systems
  - You can be disciplined, even expelled

Coming Up

Monday's lecture…

Intro to Crypto
Alice and Bob
Kerckhoffs's principle
Hashes and MACs

Next two weeks…

Applied Crypto
Randomness, Encryption
Key Exchange, Secure Channels