

Lecture 01 – The Security Mindset

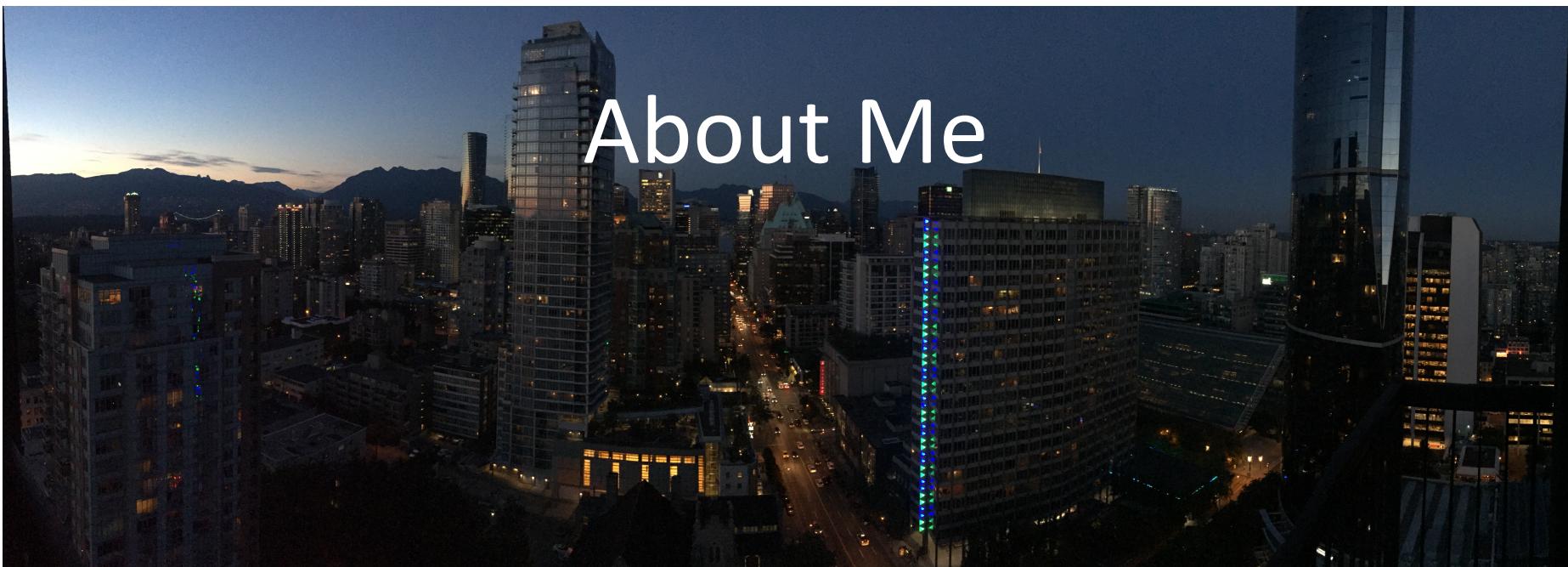
Stephen Checkoway

CS 343 – Fall 2020

Adapted from Michael Bailey's ECE 422

About Me

- Research area: Computer Security
- Some prior research
 - Voting machine security (change votes)
 - Automotive security (remote car hacks)
 - Back-scatter, whole-body X-ray scanner (weapons)
 - iSight camera (disable indicator LED while on)
 - Analysis of backdoored PRNG in TLS/IPSEC



About Me



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
- Learn to be a security-conscious citizen

Requirements

- 4 or 5 Security projects (difficult!)
- Two exams

Policies

- Attendance: not mandatory, but you should come anyway
- Late work: 3 late days
- Collaboration: Work in groups of 2 on projects
- Communication: **Don't email me!** Use Piazza
- Academic misconduct: punishment will be based on severity up to expulsion (seriously)

Examples of misconduct (nonexhaustive list)

- Claiming someone else's work as your own
- Searching for existing solutions to assignments
- Falsifying program output
- Collaborating outside your group
- Sharing code/solutions outside your group

Projects

- Work in groups of 2 (not required, *highly recommended*)
- Generally not much programming per project
- A lot of time thinking/tinkering/debugging

What is Computer Security?

- Security is a property (or more accurately a collection of properties) that hold in a given system under a given set of constraints
 - Where a system is anything from hardware, software, firmware, and information being processed, stored, and communicated.
 - and constraints define adversaries and their capabilities.
- Can also mean the measures and controls that ensure these properties
- Security is weird, as we don't *explicitly* study other properties
 - Correctness
 - Performance

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?
- Degree of access?

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.*



FAIL





Exercise

- Door lock/intercom
 - Occupant presses key which makes a tone over the intercom
 - Lock is unlocked when tone is detected over the intercom
- How can an attacker subvert this to gain access?



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?
 - What are their Capabilities? Motivations? Access?
- Risk assessment
 - What are the weaknesses of the system?
 - How likely?
- Countermeasures
 - Technical vs. nontechnical?
 - How much do they cost?

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?
 - Access?
- What kinds of attacks do we need to prevent?
(Think like the attacker!)
- Limits: Kinds of attacks we should ignore?



Assessing Risk

- 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?
- Remember: rational paranoia

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

Exercise

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

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!”

Schneier's law

- “Anyone, from the most clueless amateur to the best cryptographer, can create an algorithm that he himself can't break.”
- Replace “cryptographer” with “engineer” and “algorithm” with “system” and it still holds true



To Learn More ...

- The Security Mindset.

[https://www.schneier.com/blog/archives/2008/03/the security
mindset_1.html](https://www.schneier.com/blog/archives/2008/03/the_security_mindset_1.html)

- [https://freedom-to-tinker.com/blog/felten/security-mindset-
and-harmless-failures/](https://freedom-to-tinker.com/blog/felten/security-mindset-and-harmless-failures/)

- [https://cubist.cs.washington.edu/Security/2007/11/22/why-a-
computer-security-course-blog/](https://cubist.cs.washington.edu/Security/2007/11/22/why-a-computer-security-course-blog/)

Questions?

