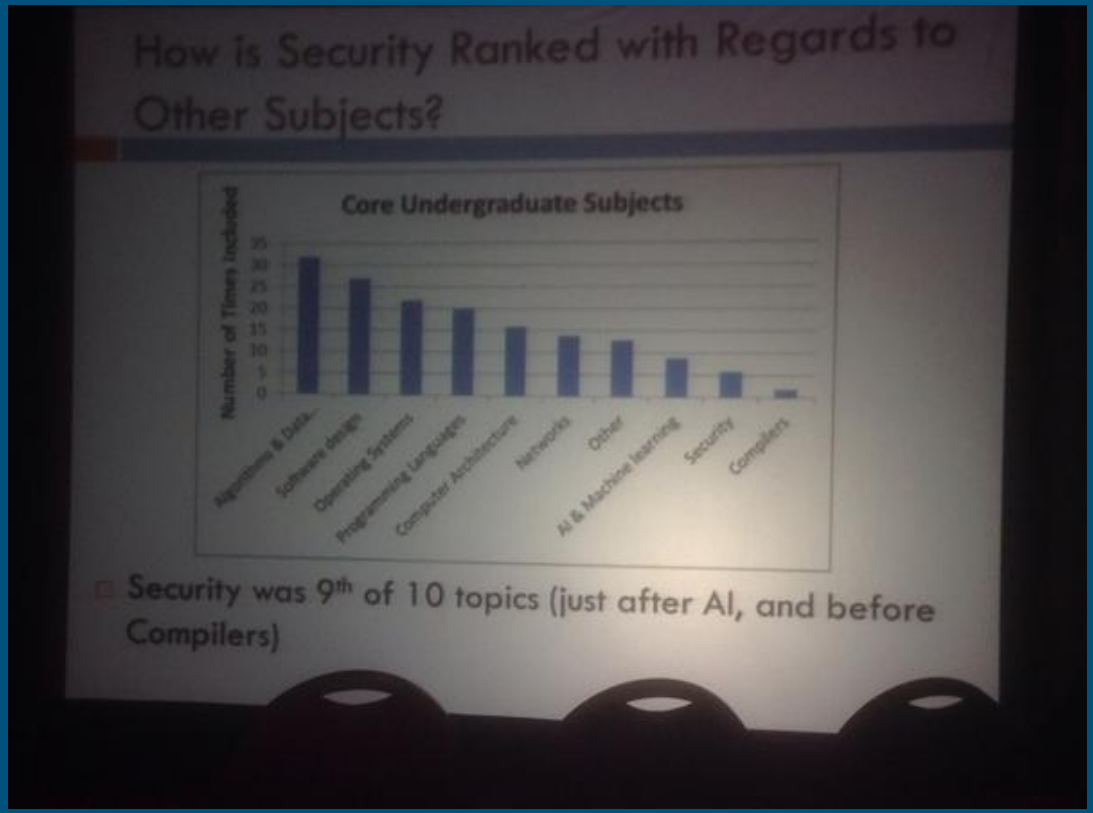
Chipping Away At The Security Education Problem

New England Security Day Spring 2016

Ming Chow (@0xmchow)
Roy Wattanasin (@wr0)



The picture that started it all for us: from HOPE X (July 2014): Sarah Zatko's How to Prevent Security Afterthought Syndrome; image from https://twitter.com/sambowne/status/490316922844872704

Wesley McGrew
@McGrewSecurity

Follow

Dunno how to teach someone all the fundamentals needed to be a good hacker other than putting them through a 4 year CS program or equivalent

RETWEETS 9 FAVORITES 10

7:21 AM - 3 Apr 2015

Tottenkoph
@tottenkoph

Follow

@McGrewSecurity I can see how a CS degree can be a good foundation, but there are some big gaps in trad CS progs if you want to be a hacker

RETWEET 1 FAVORITES 4

8:06 AM - 3 Apr 2015

InfoSec Taylor Swift
@SwiftOnSecurity

Follow

IMPORTANT: Multiple people who are graduating college have asked how they can get into InfoSec. What is your advice for people with degrees?

RETWEETS 25 FAVORITES 61

4:24 PM - 26 Apr 2015

Tottenkoph
@tottenkoph

Follow

@jth @McGrewSecurity I was surprised a group of freshmen of CS students hadn't even thought of sec as something to "get into" (job/hobby)

8:28 AM - 3 Apr 2015

Software Engineering in the Wild

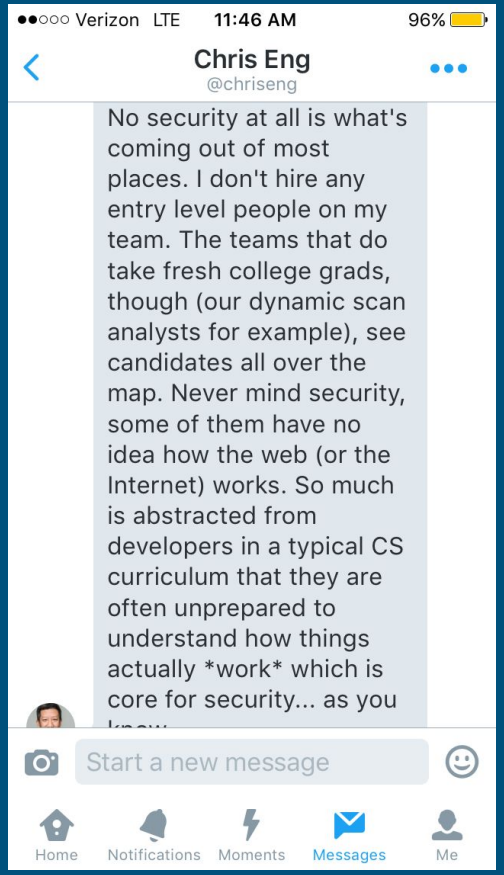
BASIC SECURITY

There are a few types of security vulnerabilities that you should be familiar with, both in terms of how to exploit them and how to harden your code against them.

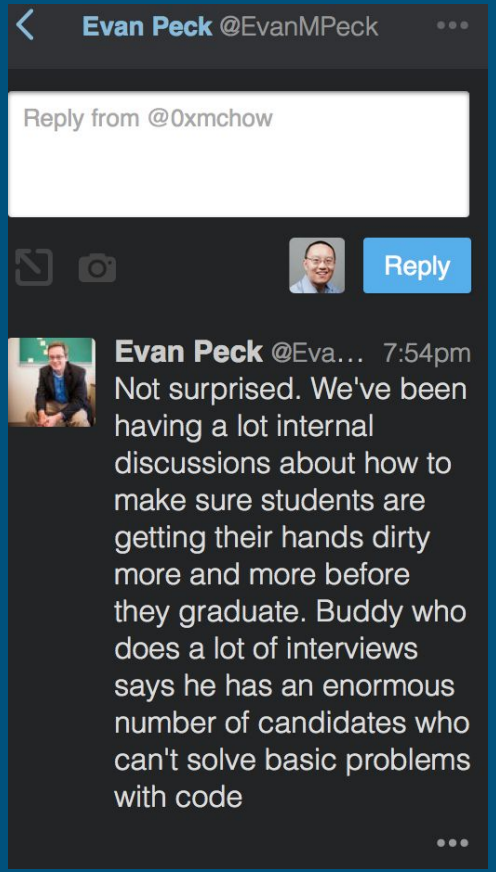
- Compromises your systems
  - Buffer overflow
  - SQL injection
- Compromises your users
  - SQL injection
  - XSS (cross-site scripting)
  - CSRF (cross-site request forgery; AKA: man-in-the-middle attack)
  - etc… (JSONP, and more)

Also be aware of PII (personally identifiable information). Don't expose or log access tokens (e.g.: Facebook, Google), e-mail addresses, or other sensitive information. Don't even store these things in plain text.

From Bill Langenberg Technical Manager, Software Engineering at TripAdvisor (guest lecture to Web Programming class at Tufts)



For your eyes only, a private message…



A comment regarding previous image...

4/7/2016
11:00 AM

Kelly Jackson Higgins
News

Top US Undergraduate Computer Science Programs Skip Cybersecurity Classes

New study reveals that none of the top 10 US university computer science and engineering program degrees requires students take a cybersecurity course.

There's the cybersecurity skills gap, but a new study shows there's also a major cybersecurity education gap -- in the top US undergraduate computer science and engineering programs.

An analysis of the top 121 US university computer science and engineering programs found that none of the top 10 requires students take a cybersecurity class for their degree in computer science, and three of the top 10 don't offer any cybersecurity courses at all. The higher-education gap in cybersecurity comes amid the backdrop of some 200,000 unfilled IT security jobs in the US, and an increasing sense of urgency for organizations to hire security talent as cybercrime and cyber espionage threats escalate.

April 7, 2016

Cybersecurity Ed: meeting the challenge

- *Challenge:* changing undergrad curriculum difficult (no knobs)
  - cybersecurity: typically, advanced undergrad elective
  - integration throughout curriculum?
- *learning from our (MA) past*: Commonwealth Information Technology Initiative (CITI)

CITI

- launched in 2000, funded by BHE
- All segments of public higher education, with industry
- "strengthen and modernize computer science and IT programs" in MA public higher ed.

From Dr. James Kurose, Assistant Director, Computer and Information Science and Engineering Directorate, NSF at Innovations Partnerships Network (IPN) Conference on December 9, 2015

A CS Curriculum's Responsibility and Obligation

- Most Computer Science curricula go through national accreditation (e.g., Accreditation Board for Engineering and Technology)
- Why is accreditation important? To assess the quality of curriculum; to ensure curriculum has basic foundations according to specific accreditation.
- One of the important outcomes of a Computer Science curriculum via ABET: **"An understanding of professional, ethical, legal, security and social issues and responsibilities"**

For Your Eyes Only

From 11/6/2011 during evaluation of Tufts' Computer Science curriculum, preliminary findings of the ABET evaluator: "There are several gaps in coverage that I have already pointed out to you and are obvious to anyone looking at a map of our coverage: > e. An understanding of professional, ethical, legal, security and social issues and responsibilities --We have part of this with EM54 (an Ethics course), **but there is little or no coverage of legal and security issues in the required curriculum.**"

Evan Peck
@EvanMPeck

Follow

@0xmchow on a side note: any good resources better integrating security into existing CS curriculum (like data structures)?

RETWEET
1

3:22 PM - 30 Dec 2014

The Key: Integration into (Most) Computer Science Courses

- Make students think #whatcouldpossiblygowrong; violate invariants, preconditions
- “Thinking like an attacker” is hard, a very different way of thinking and mindset
- Encourage students to think about security at the beginning of any project/assignment rather than being bolted on at the end
- Hands-on practice is required
- Inform them of opportunities in Security

Example: Data Structures

- The second course in most Computer Science curricula
- Discussion: the hash function for hash tables: collisions are bad but will be inevitable for simple hash functions. In the real world, hash functions are critical for security, use to verify integrity, and collisions are extremely bad (e.g.,, MD5)

Example: Web Programming

- The full--stack: HTTP, HTML5, CSS, JavaScript, server--side, data persistence using database(s)
- Build client and server, then break. Since spring 2014, students had to create “Marauder's Map”
- Issues taught: input validation, XSS, injection attacks
- Assignment: Students are paired to perform a security audit another student’s client and server.
- Example (from spring 2013): https://tuftsdev.github.io/WebProgramming/assignments/security-gjoseph/report.html

Example: Senior Capstone Project / Software Engineering

- Exercise: think of abuse cases in the specification and design phases
- Deliverable: technical risk analysis table for capstone project (in the fall)
- From @chriseng: "Undergraduate CS projects should be subjected to security testing” @sfjacob #AppSecCali2016 (/cc @0xmchow) https://twitter.com/chriseng/status/692510469442117634

Example: Game Development

- Issues taught: cheating in games, virtual economies, and abusing online games (https://tuftsdev.github.io/GameDevelopment/notes/ethics_security.html)
- Assignment: Read four accepted articles from IEEE Security & Privacy Securing Online Games issue (May/June 2009), answer five questions
- https://tuftsdev.github.io/GameDevelopment/assignments/security.html

Example: Mobile Medical Devices and Apps

- Issues taught: security and privacy of medical devices (https://mchow01.github.io/talks/SecurityMedicalDevices.pdf)
- Activities: think of security issues in the design phase
- Project 1: Build a temperature sensing device using an Arduino (hardware); iOS app to display readings
- Project 2: Build a patient monitoring device
- Guest speakers: former President of St. Elizabeth Hospital in Brighton, MA, Chief Medical Information Officer at University of California, San Francisco
- Article about our work: http://now.tufts.edu/articles/engineering-reality

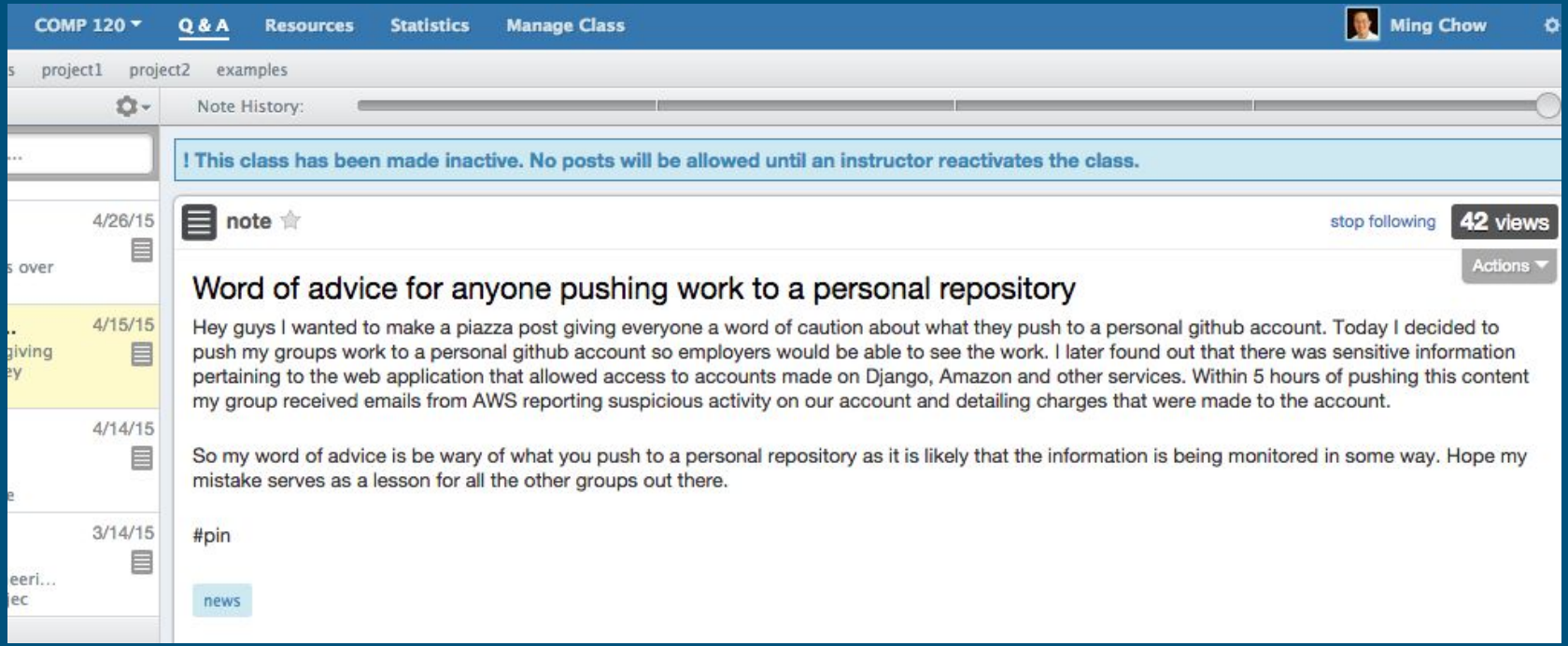
Example: Introduction to Computer Security at Tufts

- Syllabus runs the broad spectrum: network security, web security, incident handling, privacy, forensics
- Real assignments: analyze packets captured from DefCon, build an intrusion detection system (using Ruby and PacketFu)
- There is a CTF game; students play in teams
- World class guest speakers. Special thanks to Steve Christey Coley, Chris Wysopal, Peter Ballerini and his team at Putnam Investments, Kade Crockford, Gary McGraw, Vik Solem, Silicosis, Josh Abraham for their contributions over the years.

Example: Machine Structure and Assembly Language Programming

- Reverse engineering (e.g., “binary bomb”)
- Buffer overflow

Success Stories



Thanks Academic x

[redacted] Mar 9

to Ming

Hey Ming,

Just wanted to reach out an say thank you for your generous guidance and enthusiasm as a professor while I was at Tufts. Barstool Sports (my employer) recently relaunched our entire infrastructure and I was charged with the development of the mobile app and API that powers it.

The lessons learned in your courses were major influences not only in my ability to land the job, but to help build a scalable a reliable product. We just mitigated a XSS attack this morning, and I am confident that without the information I learned in your courses that the exploit would have gone unnoticed.

Thanks again Ming,

Hope all is well!

P.S. Tough year to be Arsenal or United fans. :(

Boston Globe Article

Academic x

[redacted]

5/5/10

http://www.boston.com/yourtown/news/medford/2009/08/tufts_president_among_those_li.html

So apparently I'm such a leet haxor I can hack sites without even being aware of it. Comp20 A4 taught me well?

-Mike
(pretty darn baffled - a friend found this for me today)

Never trusting user input

Academic x

[redacted] Hi Ming, Wanted to let you know that I put my Comp20 knowledge to very good u... 10:42 AM (5 hours ago)

Ming Chow Ironically, they are some of the worst developers with regards to handling so... 12:17 PM (3 hours ago)

[redacted] d be happy to have the example in the talk, as long as you remove the speci... 1:35 PM (2 hours ago)

[redacted] via cs.tufts.edu 1:42 PM (2 hours ago)

to Ming

The following would be good:

Today at my internship, I was looking at some code that sent post requests to the server and realized the input wasn't sanitized! I showed my supervisor some cross site scripting on his development server and he's now pretty frantically running around trying to determine the scope of the security holes. Hilarious that a government contractor with a cyber security department in the same building made a thing so insecure, but at least they've got a Comp20 grad to come to the rescue. I have a feeling that security might become a focus of my internship. Anyways, just wanted to say thanks for the preparation for this job, I've been using literally every part of Comp20 every day here and it's been easy so far.

...

Thanks Academic x

[redacted] Mar 9

to Ming

Hey Ming,

Just wanted to reach out an say thank you for your generous guidance and enthusiasm as a professor while I was at Tufts. Barstool Sports (my employer) recently relaunched our entire infrastructure and I was charged with the development of the mobile app and API that powers it.

The lessons learned in your courses were major influences not only in my ability to land the job, but to help build a scalable a reliable product. We just mitigated a XSS attack this morning, and I am confident that without the information I learned in your courses that the exploit would have gone unnoticed.

Thanks again Ming,

Hope all is well!

P.S. Tough year to be Arsenal or United fans. :(

[redacted]

Feb 5

to Ming

Hi Ming,

Already started reading them.

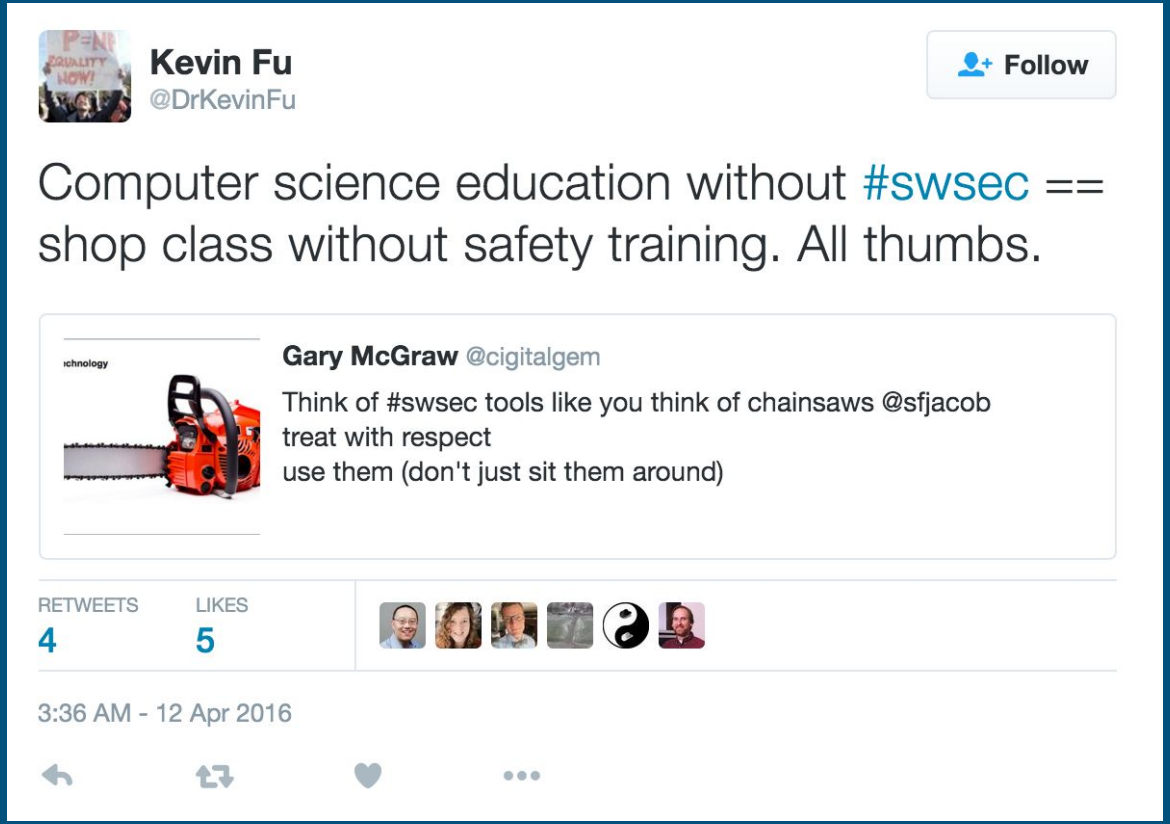
I reached out a while ago to Brian Milas from the 199 board about an internship at his startup Covered. He was impressed with me knowing what XSS was and I have a second interview coming up in a couple weeks. Thought you'd like to know hear more examples of how useful the stuff we learned in 20 is.

I have attached the instructions for the Oxford recommendation, it's due before the end of February.

Thanks!
Thomas

PS
I'll be taking Security in the fall. Can't wait

Fact to the Matter Is



The Bottom Line

- There is no excuse to not integrate security into Computer Science courses, especially systems and application--based courses.
- Inform students of the security and privacy problems and opportunities; ask students to be good citizens.
- Encourage and challenge students to develop the curiosity and mindset of the bad guy.
- Do not use only traditional teaching and learning techniques for courses. Learning how to take tests isn’t helping.
- Provide mentorship and networking opportunities.

Chris Eng
@chriseng

Follow

"We can't address the talent gap without retrofitting the CS degree." Update textbooks, educate professors, etc. @sfjacob #AppSecCali2016

RETWEETS 5 LIKES 5

4:50 PM - 27 Jan 2016

Tad Taylor @tad_taylor · Jan 27
@chriseng @cigitalgem @sfjacob CS degree needs to bring logic, philosophy, etc. to teach people to think & communicate. #swsec = thinking

1

Only the Tip of the Iceberg

Only the Tip of the Iceberg (continued)

“The title “engineer” is cheapened by the tech industry.

Recent years have seen prominent failures in software. Massive data breaches at Target, Home Depot, BlueCross BlueShield, Anthem, Harvard University, LastPass, and Ashley Madison only scratch the surface of the cybersecurity issues posed by today’s computer systems. The Volkswagen diesel-emissions exploit was caused by a software failing, even if it seems to have been engineered, as it were, deliberately.”

Addendum: Theranos

Source: http://www.theatlantic.com/technology/archive/2015/11/programmers-should-not-call-themselves-engineers/414271/?single_page=true

NCEES
advancing licensure for engineers and surveyors

NCEES Principles and Practice of Engineering Examination
Software Engineering Exam Specifications

Effective Beginning with the April 2013 Examinations

- The exam is an 8-hour open-book exam. It contains 40 multiple-choice questions in the 4-hour morning session, and 40 multiple-choice questions in the 4-hour afternoon session. Examinee works all questions.
- The exam uses both the International System of units (SI) and the US Customary System (USCS).
- The exam is developed with questions that will require a variety of approaches and methodologies, including design, analysis, and application.
- The knowledge areas specified as examples of kinds of knowledge are not exclusive or exhaustive categories.

decisions under uncertainty)

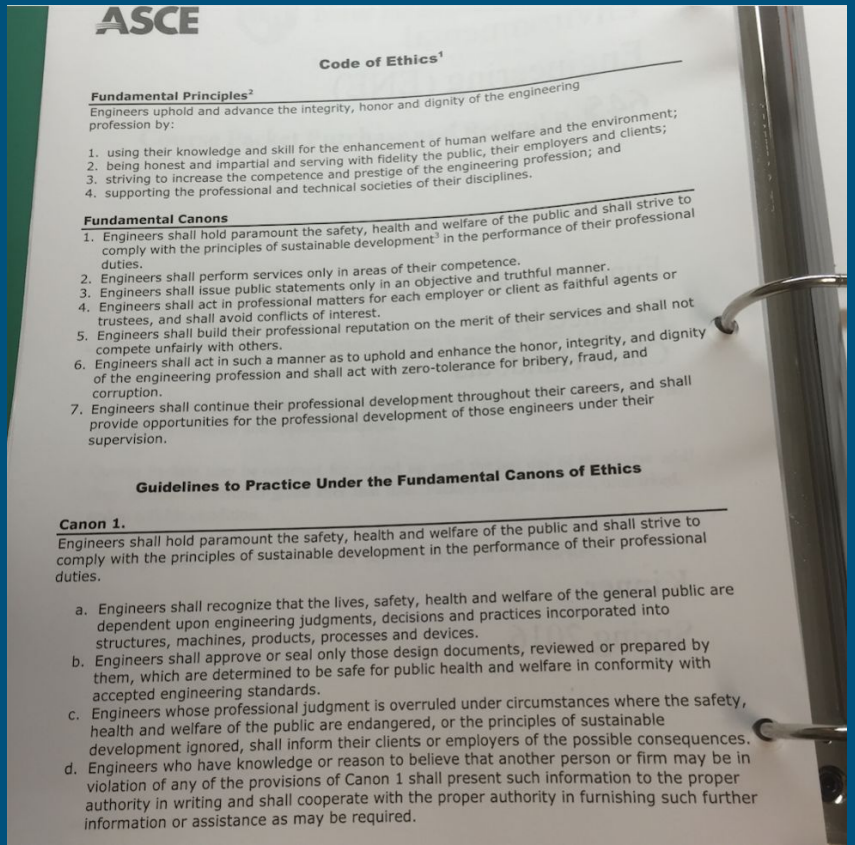
VIII. Quality Assurance **6**

A. Software quality fundamentals (e.g., organizational role, value and cost of quality, models and quality characteristics, software quality improvement)
B. Software quality management processes and systems (e.g., product assurance, process assurance, quality analysis and evaluation
C. Software quality techniques (e.g., reviews, audits, software quality requirements, defect characterization, software quality measurement, software quality tools)

IX. Safety, Security, and Privacy **12**

A. Basic concepts (e.g., security versus privacy, intellectual property, confidentiality, integrity, availability, Common Criteria, component criticality)
B. Secure architecture and design (e.g., secure communications, disaster recovery, encryption, patterns and anti-patterns, infrastructure and environment planning)
C. Secure coding (e.g., secure subsets, encryption and keying, numerical precision, accuracy and errors)
D. Human–computer interface design (e.g., use of shape and color, response time, system navigation, consistency, error messages)
E. Safety issues (e.g., hazard analysis, failure analysis, fault-tolerant design, fail-safe design, fault-recovery)
F. Identity, authentication, and authorization (e.g., biometrics, password strength)
G. Threat analysis and remediation (e.g., spoofing, tampering, repudiation, information disclosure, denial of service, elevation of privilege, assurance cases, security audits)
H. Security testing (e.g., penetration testing, intrusion detection, fuzz testing, fault injection)

A Uniform Code of Ethics to Abide By?



ASCE

Code of Ethics[1]

Fundamental Principles[2]

Engineers uphold and advance the integrity, honor and dignity of the engineering profession by:

1. using their knowledge and skill for the enhancement of human welfare and the environment;
2. being honest and impartial and serving with fidelity the public, their employers and clients;
3. striving to increase the competence and prestige of the engineering profession; and
4. supporting the professional and technical societies of their disciplines.

Fundamental Canons

1. Engineers shall hold paramount the safety, health and welfare of the public and shall strive to comply with the principles of sustainable development[3] in the performance of their professional duties.
2. Engineers shall perform services only in areas of their competence.
3. Engineers shall issue public statements only in an objective and truthful manner.
4. Engineers shall act in professional matters for each employer or client as faithful agents or trustees, and shall avoid conflicts of interest.
5. Engineers shall build their professional reputation on the merit of their services and shall not compete unfairly with others.
6. Engineers shall act in such a manner as to uphold and enhance the honor, integrity, and dignity of the engineering profession and shall act with zero-tolerance for bribery, fraud, and corruption.
7. Engineers shall continue their professional development throughout their careers, and shall provide opportunities for the professional development of those engineers under their supervision.

Guidelines to Practice Under the Fundamental Canons of Ethics

Canon 1.

Engineers shall hold paramount the safety, health and welfare of the public and shall strive to comply with the principles of sustainable development in the performance of their professional duties.

a. Engineers shall recognize that the lives, safety, health and welfare of the general public are dependent upon engineering judgments, decisions and practices incorporated into structures, machines, products, processes and devices.
b. Engineers shall approve or seal only those design documents, reviewed or prepared by them, which are determined to be safe for public health and welfare in conformity with accepted engineering standards.
c. Engineers whose professional judgment is overruled under circumstances where the safety, health and welfare of the public are endangered, or the principles of sustainable development ignored, shall inform their clients or employers of the possible consequences.
d. Engineers who have knowledge or reason to believe that another person or firm may be in violation of any of the provisions of Canon 1 shall present such information to the proper authority in writing and shall cooperate with the proper authority in furnishing such further information or assistance as may be required.

Resources and References

- “[HOPE X] How to Prevent Security Afterthought Syndrome” https://www.youtube.com/watch?v=iLiQqii0c9E
- http://www.theatlantic.com/technology/archive/2015/11/programmers-should-not-call-themselves-engineers/414271/?single_page=true
- http://www.theatlantic.com/technology/archive/2015/12/the-moral-failure-of-computer-science/420012/
- http://www.massinsight.com/wp-content/uploads/2015/12/IPN-Conf-2015_Kurose.pdf
- http://www.darkreading.com/vulnerabilities---threats/top-us-undergraduate-computer-science-programs-skip-cybersecurity-classes/d/d-id/1325024
- https://tuftsdev.github.io/WebProgramming/notes/blangenberg.pdf
- http://www.scmagazine.com/updated-cybersecurity-being-overlooked-by-american-universities-report/article/488233/
- http://www.irongeek.com/i.php?page=videos/teaching-hacking-at-college-sam-bowne
- https://www.defcon.org/images/defcon-15/dc15-presentations/dc-15-bownpdf
- http://www.slideshare.net/cchardin/bsides-las-vegas-caroline-d-hardin-on-hacking-education
- https://www.defcon.org/images/defcon-17/dc-17-presentations/defcon-17-jon_kibler-mike_cooper-hack_the_textbook.pdf
- https://www.defcon.org/html/defcon-22/dc-22-speakers.html#Erven
- https://cdn.ncees.org/wp-content/uploads/2012/11/SWE-Apr-2013.pdf