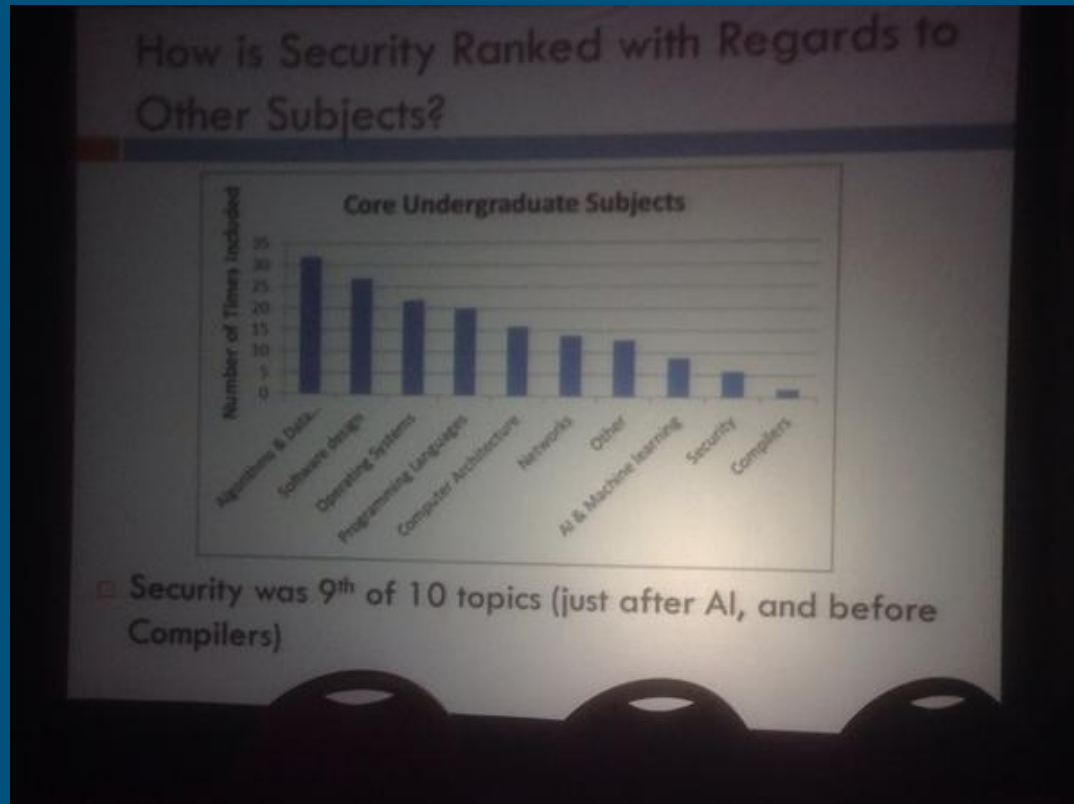


Chipping Away At The Security Education Problem

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](https://twitter.com/sambowne/status/490316922844872704); image from <https://twitter.com/sambowne/status/490316922844872704>

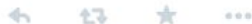


Wesley McGrew

@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

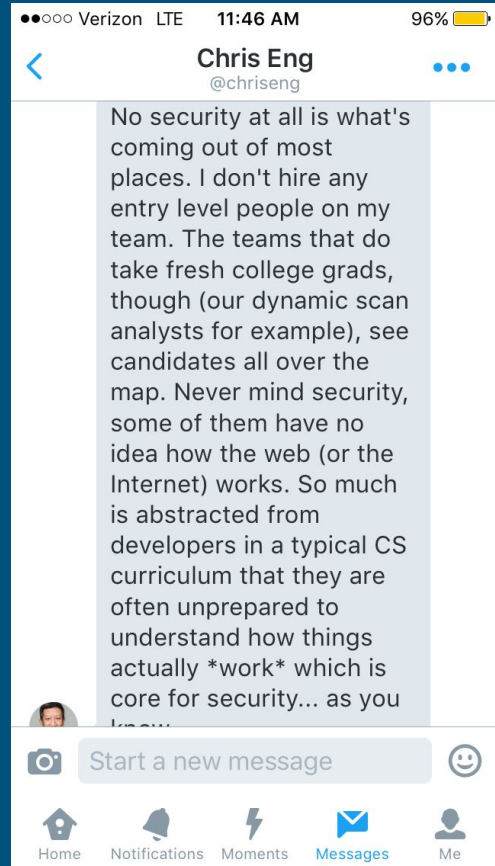
BASIC SECURITY

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.

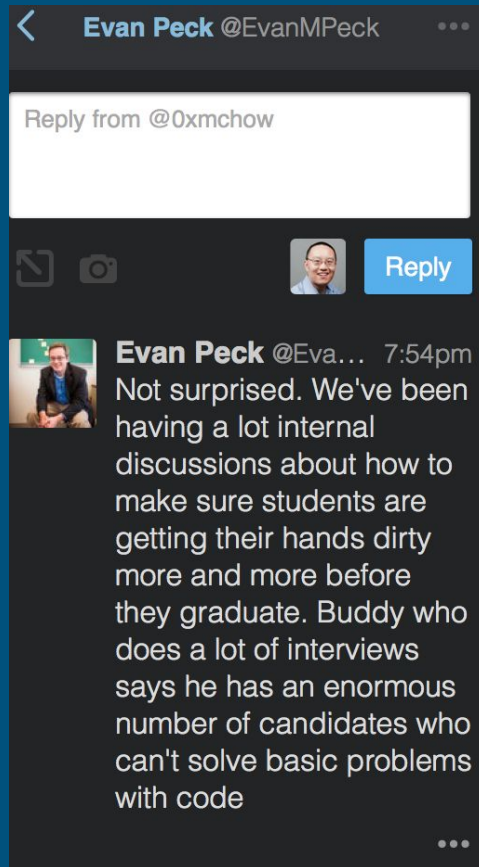
- Compromises your systems
 - Buffer overflow
 - SQL injection
- Compromises your users
 - SQL injection
 - XSS (cross-site scripting)
 - CSRF (cross-site request forgery; AKA: man-in-the-middle attack)
 - etc... (JSONP, and more)

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...

[Home](#) [News & Commentary](#) [Authors](#) [Slideshows](#) [Video](#) [Radio](#) [Reports](#) [White Papers](#) [Events](#)

[ANALYTICS](#) [ATTACKS / BREACHES](#) [APP SEC](#) [CAREERS & PEOPLE](#) [CLOUD](#) [ENDPOINT](#) [IoT](#) [MOBILE](#) [OPERATIONS](#)

VULNERABILITIES / THREATS

4/7/2016
11:00 AM



Kelly Jackson
Higgins
News

Connect Directly



0 COMMENTS
[COMMENT NOW](#)

[Login](#)



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

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

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

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

The screenshot shows a web interface for a course named "COMP 120". The top navigation bar includes links for "Q & A", "Resources", "Statistics", and "Manage Class". The user profile "Ming Chow" is visible in the top right. Below the navigation bar, there are tabs for "project1", "project2", and "examples". A "Note History" section with a scrollbar is present. A blue warning banner states: "This class has been made inactive. No posts will be allowed until an instructor reactivates the class." The main content area displays a post titled "Word of advice for anyone pushing work to a personal repository" by a user named "note". The post has 42 views and a "stop following" button. The text of the post discusses a caution about pushing work to a personal GitHub account, mentioning sensitive information and AWS activity. A "#pin" tag and a "news" button are at the bottom of the post. A left sidebar shows a list of dates and document icons, with "4/15/15" highlighted in yellow.

COMP 120 ▾ Q & A Resources Statistics Manage Class

s project1 project2 examples

Note History: [Progress Bar]

! This class has been made inactive. No posts will be allowed until an instructor reactivates the class.

note ☆ stop following 42 views Actions ▾

Word of advice for anyone pushing work to a personal repository

Hey guys I wanted to make a piazza post giving everyone a word of caution about what they push to a personal github account. Today I decided to push my groups work to a personal github account so employers would be able to see the work. I later found out that there was sensitive information pertaining to the web application that allowed access to accounts made on Django, Amazon and other services. Within 5 hours of pushing this content my group received emails from AWS reporting suspicious activity on our account and detailing charges that were made to the account.

So my word of advice is be wary of what you push to a personal repository as it is likely that the information is being monitored in some way. Hope my mistake serves as a lesson for all the other groups out there.

#pin

news

4/26/15
s over
4/15/15
giving
ey
4/14/15
e
3/14/15
eeri...
ec

Thanks



Academic x



Mar 9 ☆



to Ming

Hey Ming,

Just wanted to reach out and 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 and 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



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



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

...



🔒 to Ming ▾

📎 Feb 5 ☆ ↩ ▾

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



Kevin Fu
@DrKevinFu

 Follow

Computer science education without [#swsec](#) == shop class without safety training. All thumbs.

technology



Gary McGraw @cigitalgem

Think of [#swsec](#) tools like you think of chainsaws @sfjacob
treat with respect
use them (don't just sit them around)

RETWEETS

4

LIKES

5



3:36 AM - 12 Apr 2016



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

 SUBSCRIBE SEARCH MENU

Programmers: Stop Calling Yourself Engineers



Programmers: Stop Calling Yourself Engineers

It undermines a long tradition of designing and building infrastructure in the public interest.

32k



TEXT SIZE



IAN BOGOST | NOV 5, 2015 | TECHNOLOGY

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 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

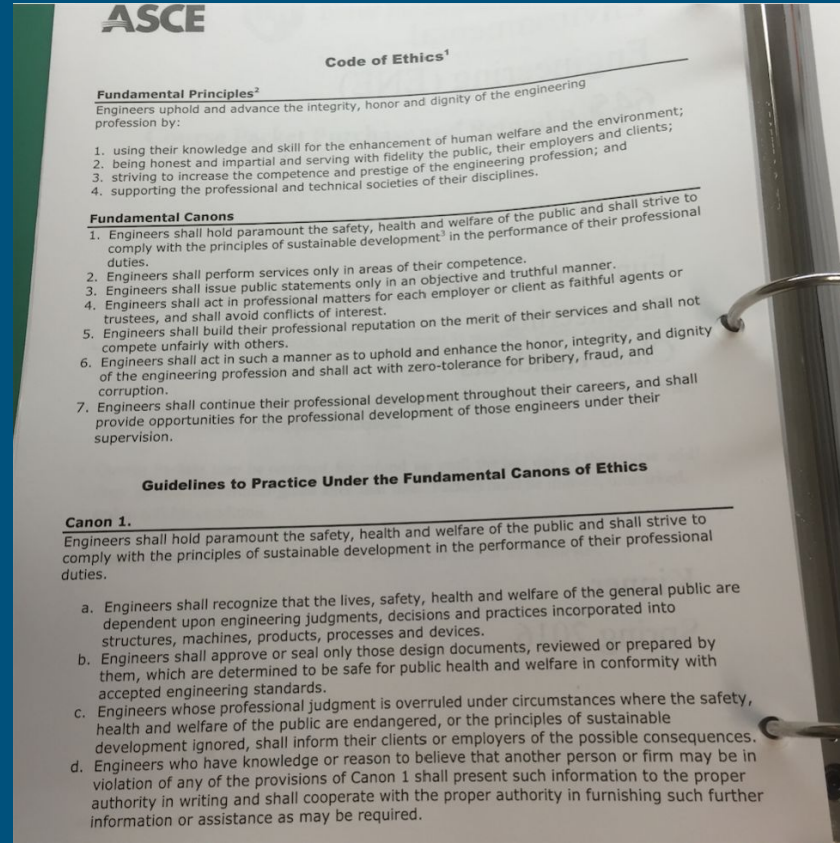
- A. Software quality fundamentals (e.g., organizational role, value and cost of quality, models and quality characteristics, software quality improvement)
- B. Software quality management processes and systems (e.g., product assurance, process assurance, quality analysis and evaluation)
- C. Software 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?



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>