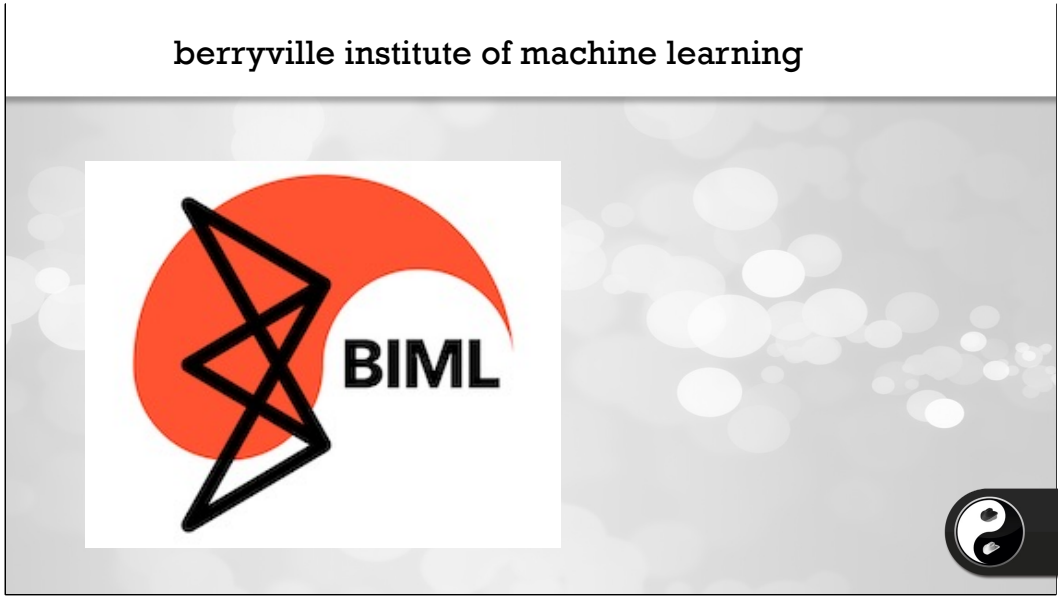
Please live tweet this talk

For more see https://garymcgraw.com

where I'm coming from

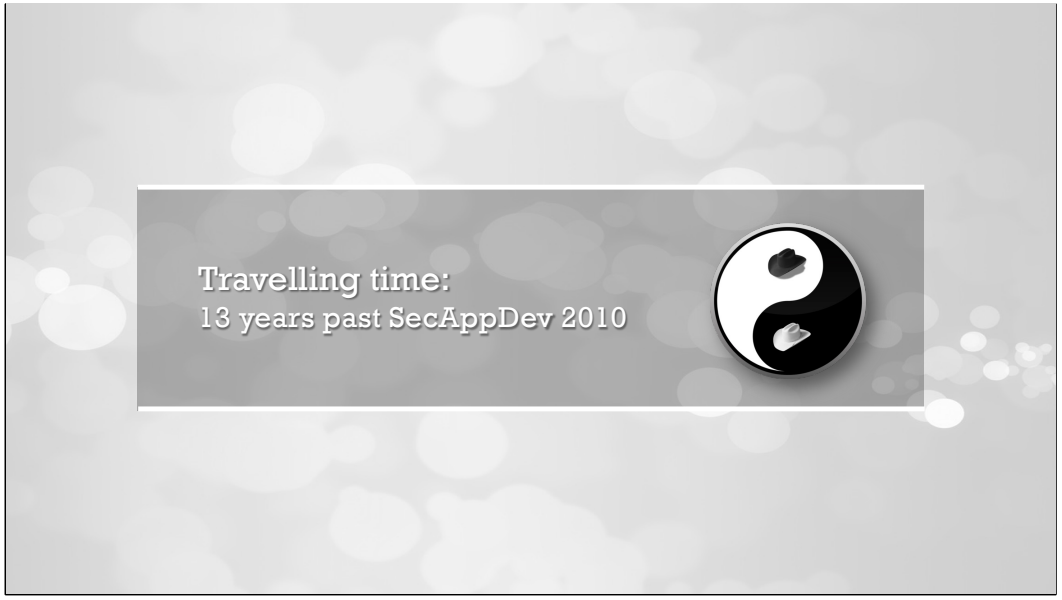


Technology	Invented the field of software security (12 books) Cigital to Synopsys (500 people)
Music	Funny faces while playing the violin
Life	Country Roads Fiction reader, Art collector, Craft cocktail maker, Cook Solstice parties



Talk TODAY at 5:30

https://berryvilleiml.com



Slides for my 2012 TROOPERS talk are here
https://troopers.de/media/filer_public/d9/9c/d99c07da-1367-4d60-9b54-cb5fe4226979/tr12_keynote_day01_mcgraw_bug_parades_zombies_and_the_bsimm.pdf

42 things in six sets

- SIX software security zombies
- TEN flaws
- SEVEN myths
- SEVEN startup lessons
- FOUR tribes
- SEVEN things I learned

- ONE BONUS THING

How do you boil down a decade of work?

See pointers to original materials at the end!

Zombies https://www.informit.com/articles/article.aspx?p=1739924

six software security zombies

ZOMBIE

Network security FAIL

More code more bugs

SDLC integration

Bugs and flaws

Badness-ometers

Fix the software

Finding and FIXING faster

CSD http://bit.ly/ieee-CSD-gem

review: on bugs and flaws

IMPLEMENTATION BUGS

- Buffer overflow
  - String format
  - One-stage attacks
- Race conditions
  - TOCTOU (time of check to time of use)
- Unsafe environment variables
- Unsafe system calls
  - System()
- Untrusted input problems

50%

ARCHITECTURE FLAWS

- Misuse of cryptography
- Compartmentalization problems in design
- Privileged block protection failure (DoPrivileged())
- Catastrophic security failure (fragility)
- Type safety confusion error
- Insecure auditing
- Broken or illogical access control (RBAC over tiers)
- Method over-riding problems (subclass issues)
- Signing too much code

50%

Two kinds of software defect

Sometimes fixing the architecture (at Google for example) can eradicate jillions of FLAWS (XXS made much harder)

The easiest flaw in the world: “FORGOT TO AUTHENTICATE USER”

ten flaws (not bugs)

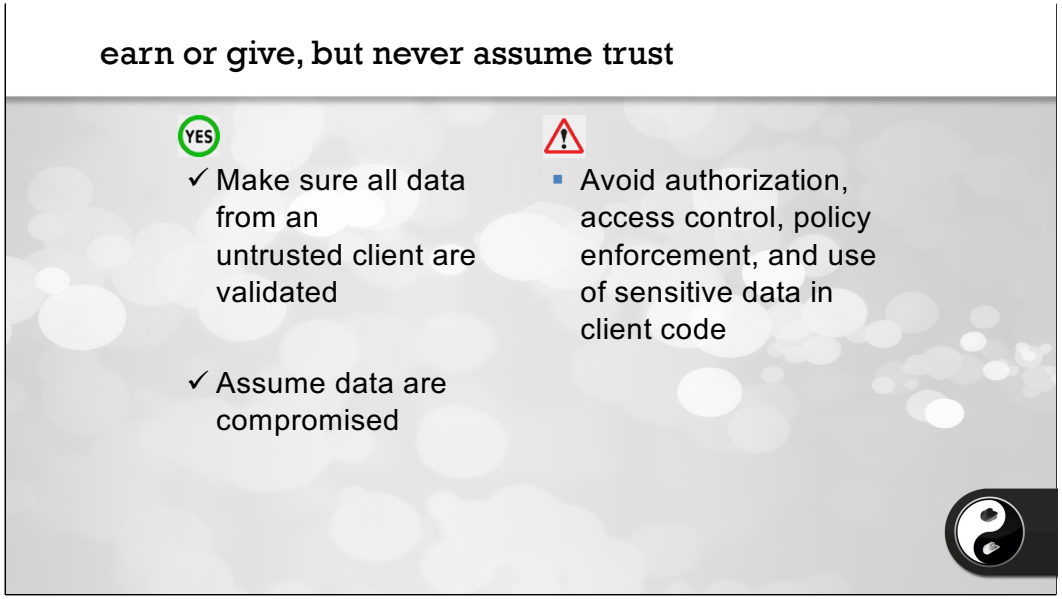
IEEE CENTER FOR SECURE DESIGN

- Earn or give, but never assume, trust
- Use an authentication mechanism that cannot be bypassed or tampered with
- Authorize after you authenticate
- Strictly separate data and control instructions, and never process control instructions received from untrusted sources
- Define an approach that ensures all data are explicitly validated
- Use cryptography correctly
- Identify sensitive data and how they should be handled
- Always consider the users
- Understand how integrating external components changes your attack surface
- Be flexible when considering future changes to objects and actors

Two kinds of software defect

Sometimes fixing the architecture (at Google for example) can eradicate jillions of FLAWS (XXS made much harder)

The easiest flaw in the world: “FORGOT TO AUTHENTICATE USER”



ML** WHERE DID THOSE DATA COME FROM?

===

60% of machine learning risks are related to data issues. Public data can be biased and sometimes even intentionally poisoned.

MACHINE LEARNING SYSTEMS don't have a good answer to this set of risks yet

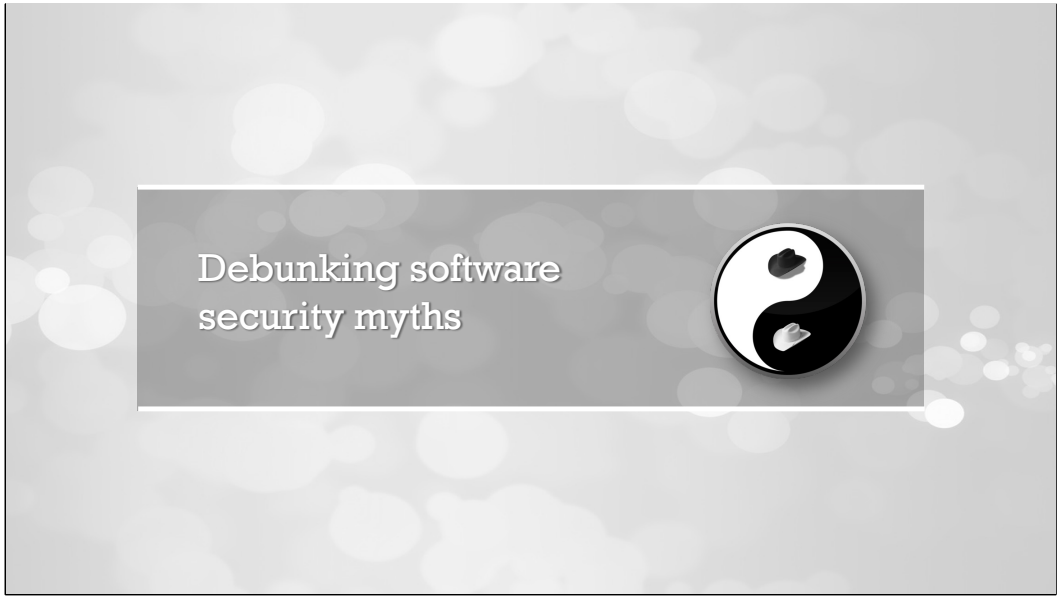
WHO IS CALLING YOUR API??

===

Most early android escalation of privilege (oh, sorry, "jailbreaking") flaws followed policy #1. System services assumed the information or messages they'd get were from authorized sources.

===

Delivery people being allowed inside. I even see this happen on accident during engagements when I'm in NYC. They have enough messengers there when I arrive security tends to just show me through as I use a messenger bag.



Seven Myths of Swsec http://bit.ly/swsec-myths

seven myths of software security

- Perimeter security can secure your applications
- A tool is all you need for software security
- Penetration testing solves everything
- Software security is a cryptography problem
- Software security is only about finding bugs in your code
- Software security should be solved by developers
- Only high-risk applications need to be secured

cryptography is magic

- The liberal application of “magic crypto fairy dust” does not address defects.

Startup Lessons https://www.informit.com/articles/article.aspx?p=1403996

startup lessons

Think and write
Build a network
Follow the Categorical Imperative
Achieve the Buddha calm
Develop a rhythm
Follow your passion
Build great stuff

It's not all just technology

achieve the Buddha calm

- Don’t panic.
- Leadership is key. Seek adult supervision. Set realistic goals with (not for) everyone.
- Develop and use metrics.
  - EBITDA
  - Open book management

CISO Report http://bit.ly/CISO-4tribes

four CISO tribes and where to find them

- Tribe 1: Security as Enabler
- Tribe 2: Security as Technology
- Tribe 3: Security as Compliance
- Tribe 4: Security as Cost Center

This is a joke

security as enabler

- Evolved from compliance to commitment (even the Board)
- Business-focused approach means LoB participation

- Balanced staff
- Look like senior executive peers
- Get in front of standards by which they will be judged

Zappa and Boyle and McGraw at Shmoocon
https://www.youtube.com/watch?v=YCWoQ2ru4mA

seven things I (may) have learned in 21 years

1. Passion matters.
2. So does a good rhythm section.
3. Practice, then practice some more.
4. Write original music.
5. Find the calm.
6. Give back.
7. Know your audience.

T.C. Boyle

Frank Zappa

A retrospective for Shmoocon

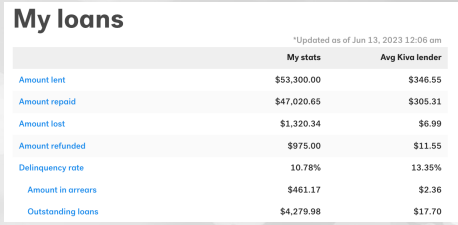
give back

We are animals and we are made in this way and this is how we behave. I'm just kind of fascinated by how we can deny that we are animals and what our impact on the other animals is like, and how quixotic we can be in trying to assess what we've done in trying to correct it.

you may make your own luck, but SHARE IT

- Give back to others in your field.
- Give back to your community, especially those in need.

- No money? Give your time.
- We're all monkeys on this planet together.



My loans

*Updated as of Jun 13, 2023 12:06 am

	My stats	Avg Kiva lender
Amount lent	$53,300.00	$346.55
Amount repaid	$47,020.65	$305.31
Amount lost	$1,320.34	$6.99
Amount refunded	$975.00	$11.55
Delinquency rate	10.78%	13.35%
Amount in arrears	$461.17	$2.36
Outstanding loans	$4,279.98	$17.70

KIVA

Kiva http://bit.ly/cigitalgem-kiva

Detail the details

Thing 42: Pointers to the details

1. Zombies https://www.informit.com/articles/article.aspx?p=1739924
2. CSD http://bit.ly/ieee-CSD-gem
3. Seven Myths of Swsec http://bit.ly/swsec-myths
4. Startup Lessons https://www.informit.com/articles/article.aspx?p=1403996
5. CISO Report http://bit.ly/CISO-4tribes
6. Zappa and Boyle and McGraw at Shmoocon https://www.youtube.com/watch?v=YCWoQ2ru4mA
7. Kiva http://bit.ly/cigitalgem-kiva

Learning more

Come to the Machine Learning Security talk too! See what I have been up to since "retiring."

https://berryvilleiml.com/results/ara.pdf