Prof. Philip Koopman

EMBEDDED SOFTWARE SECURITY, SAFETY & QUALITY

Why it matters. What to do about it.

18-642 Overview

Fall 2021

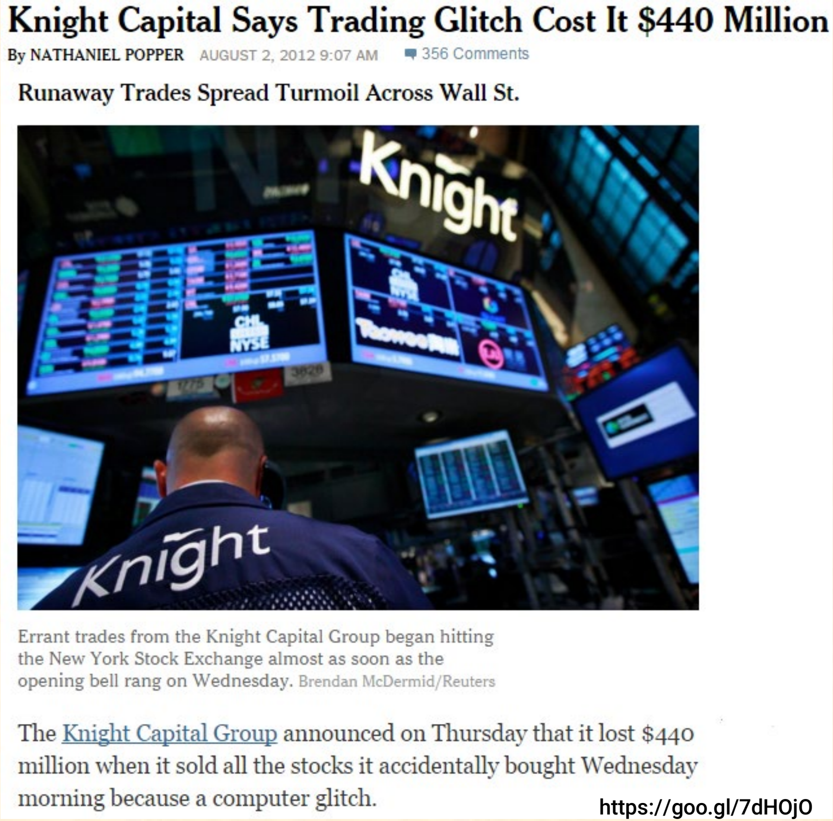
@PhilKoopman

Electrical & Computer ENGINEERING

Carnegie Mellon University

One Software Mistake Is All It Takes

- Bad software can tarnish the brand…or kill the company



Knight Capital Says Trading Glitch Cost It $440 Million

By NATHANIEL POPPER AUGUST 2, 2012 9:07 AM 356 Comments

Runaway Trades Spread Turmoil Across Wall St.

Errant trades from the Knight Capital Group began hitting the New York Stock Exchange almost as soon as the opening bell rang on Wednesday. Brendan McDermid/Reuters

The Knight Capital Group announced on Thursday that it lost $440 million when it sold all the stocks it accidentally bought Wednesday morning because a computer glitch.

https://goo.gl/7dHOjO

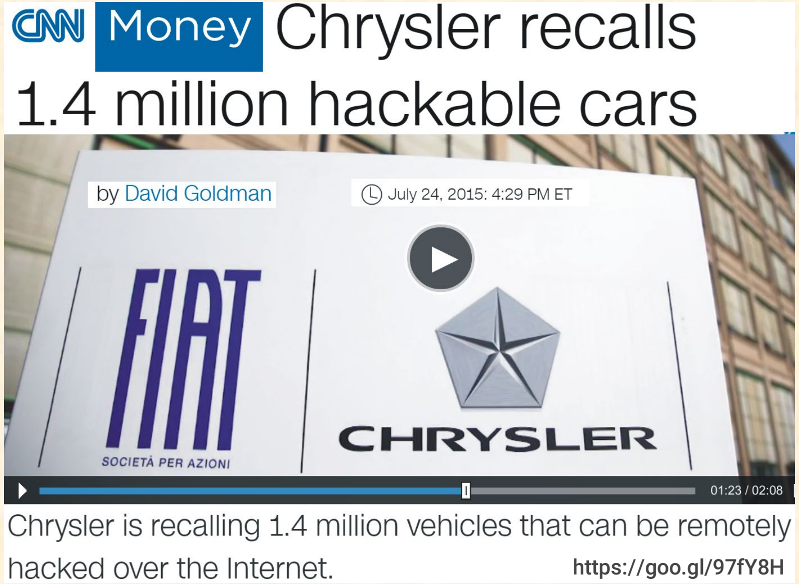
The scandal cost Martin Winterkorn his position as chief executive of VW Photo: AP/Richard Drew

https://goo.gl/T96ezC

Will “Diesel-Gate” Kill VW?

Overview

- Software quality problems are pervasive
  - Are you going to wait until you're on CNN to do something about it?
- Your company lives or dies by its software quality
  - Software is a core competency … … whether you like it or not
  - Embedded software requires unique skills & technical approaches



CNN Money Chrysler recalls 1.4 million hackable cars

by David Goldman July 24, 2015: 4:29 PM ET

Chrysler is recalling 1.4 million vehicles that can be remotely hacked over the Internet.

https://goo.gl/97fY8H

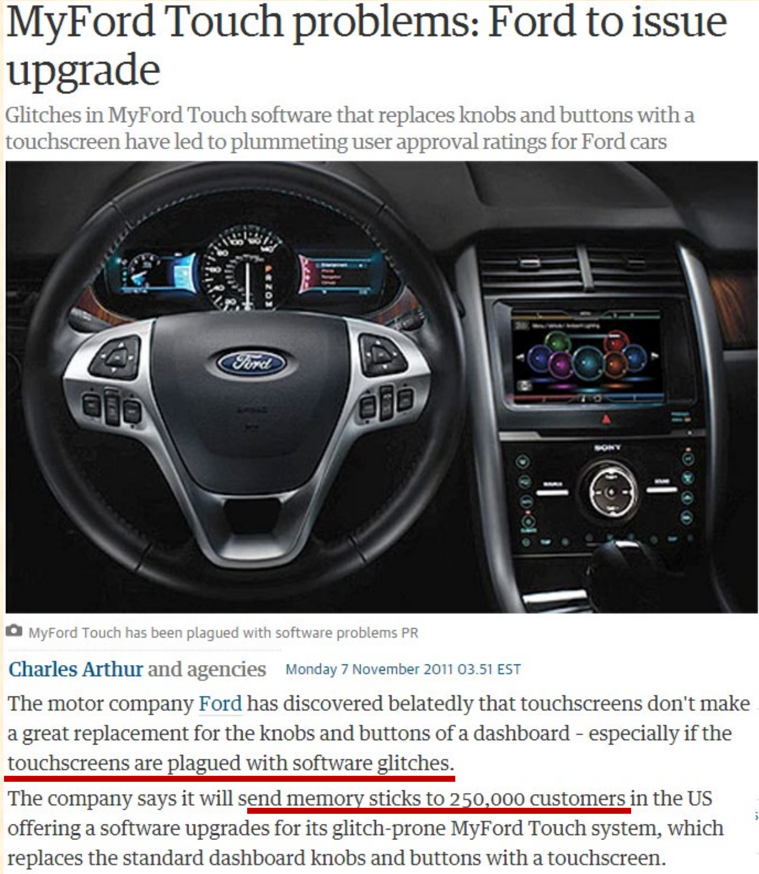
- More product-level testing won't make this problem go away
  - Need good practices, development process, development skills
- Get serious about software quality
  - Daily practices, process support, training, metrics

Embedded Software Is Challenging

- Customers expect “perfect” embedded SW
  - Everyday desktop quality software isn’t good enough
  - Bugs can lead to class action lawsuits
  - Upgrades can be painful to deploy

- Significant technical challenges
  - Limited hardware resources
  - Real-time operation
  - Interaction with system-specific sensors and actuators

- Most embedded software is Mission Critical
  - Safety: someone gets killed or injured
  - Mission Critical: failure results in unacceptable loss (money, business,…)



MyFord Touch problems: Ford to issue upgrade

Glitches in MyFord Touch software that replaces knobs and buttons with a touchscreen have led to plummeting user approval ratings for Ford cars

MyFord Touch has been plagued with software problems PR

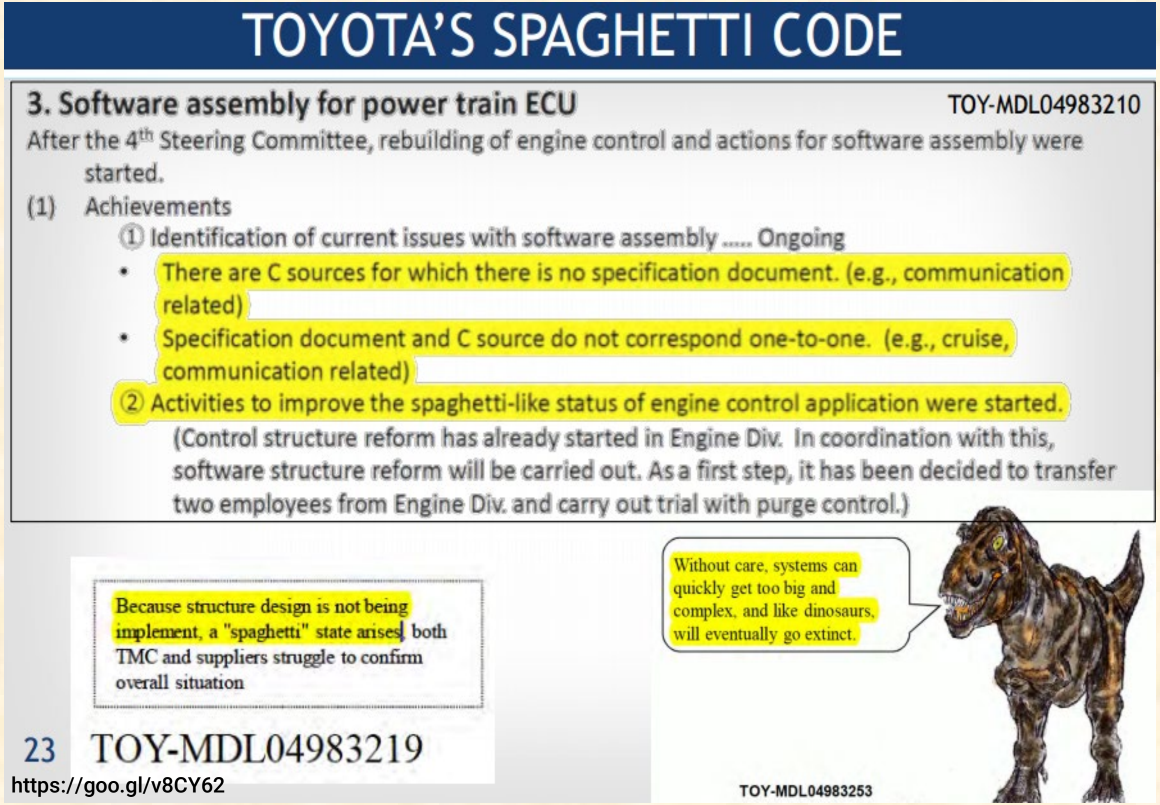
Charles Arthur and agencies Monday 7 November 2011 03.51 EST

The motor company Ford has discovered belatedly that touchscreens don't make a great replacement for the knobs and buttons of a dashboard - especially if the touchscreens are plagued with software glitches.

The company says it will send memory sticks to 250,000 customers in the US offering a software upgrades for its glitch-prone MyFord Touch system, which replaces the standard dashboard knobs and buttons with a touchscreen.

https://goo.gl/4bS5rd

Some Code Is Pervasively Bad



TOYOTA'S SPAGHETTI CODE

3. Software assembly for power train ECU TOY-MDL04983210

After the 4th Steering Committee, rebuilding of engine control and actions for software assembly were started.

(1) Achievements

① Identification of current issues with software assembly ..... Ongoing

- There are C sources for which there is no specification document. (e.g., communication related)
- Specification document and C source do not correspond one-to-one. (e.g., cruise, communication related)

② Activities to improve the spaghetti-like status of engine control application were started.

(Control structure reform has already started in Engine Div. In coordination with this, software structure reform will be carried out. As a first step, it has been decided to transfer two employees from Engine Div. and carry out trial with purge control.)

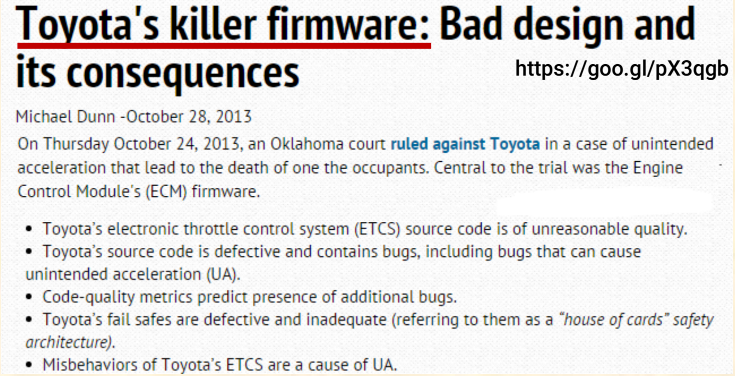
Because structure design is not being implement, a "spaghetti" state arises both TMC and suppliers struggle to confirm overall situation

23 TOY-MDL04983219

Without care, systems can quickly get too big and complex, and like dinosaurs, will eventually go extinct.

TOY-MDL04983253

https://goo.gl/v8CY62



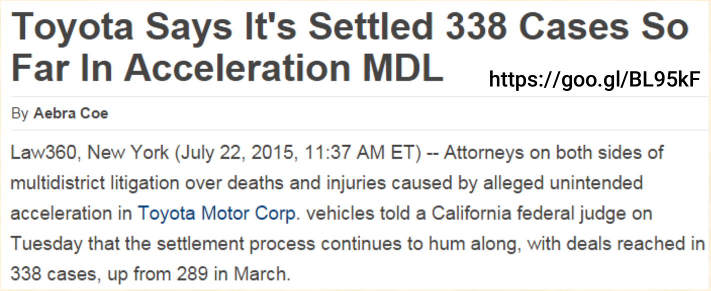
Toyota's killer firmware: Bad design and its consequences

https://goo.gl/pX3qgb

Michael Dunn -October 28, 2013

On Thursday October 24, 2013, an Oklahoma court **ruled against Toyota** in a case of unintended acceleration that lead to the death of one the occupants. Central to the trial was the Engine Control Module's (ECM) firmware.

- Toyota's electronic throttle control system (ETCS) source code is of unreasonable quality.
- Toyota's source code is defective and contains bugs, including bugs that can cause unintended acceleration (UA).
- Code-quality metrics predict presence of additional bugs.
- Toyota's fail safes are defective and inadequate (referring to them as a *"house of cards" safety architecture*).
- Misbehaviors of Toyota's ETCS are a cause of UA.



Toyota Says It's Settled 338 Cases So Far In Acceleration MDL

https://goo.gl/BL95kF

By **Aebra Coe**

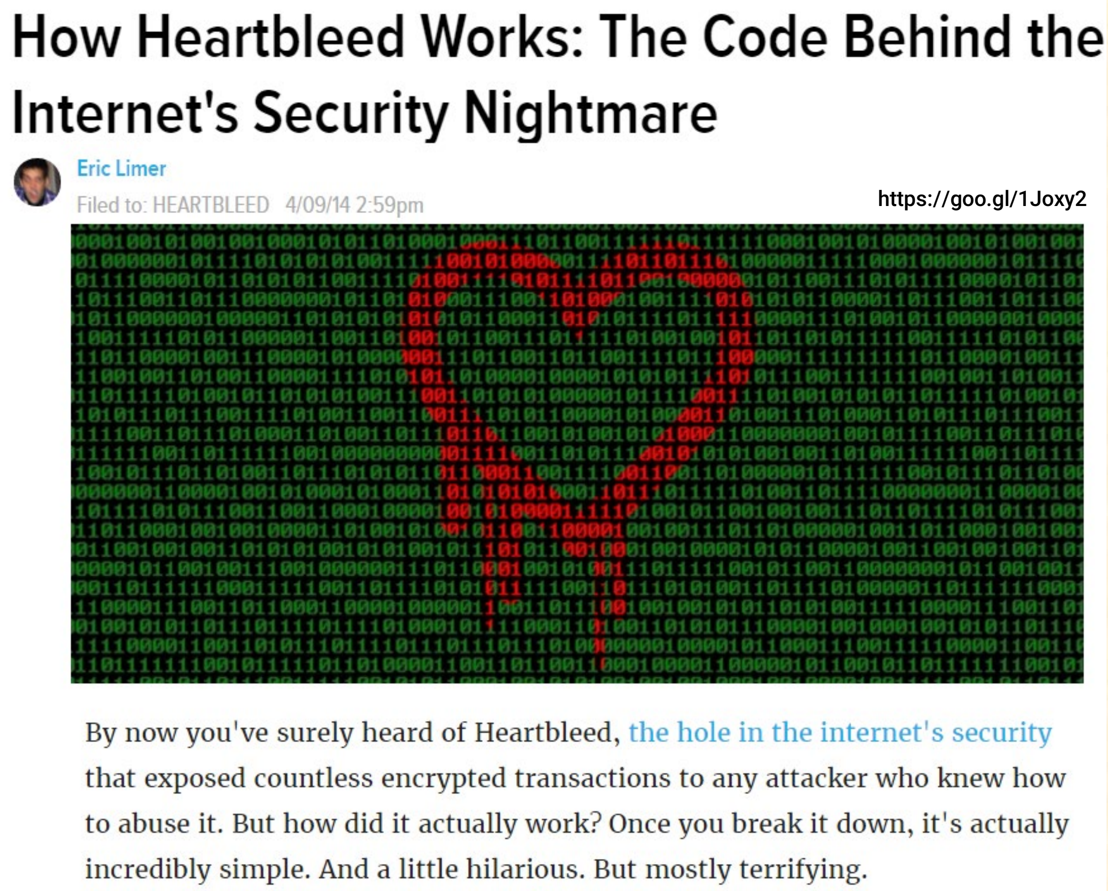
Law360, New York (July 22, 2015, 11:37 AM ET) -- Attorneys on both sides of multidistrict litigation over deaths and injuries caused by alleged unintended acceleration in Toyota Motor Corp. vehicles told a California federal judge on Tuesday that the settlement process continues to hum along, with deals reached in 338 cases, up from 289 in March.

But It Only Takes One Bad Line of Code

- This is the bad line of code for Heartbleed:

```
memcpy(bp,pl,payload);
```

  - Classic buffer overflow vulnerability
    - Copies “payload” bytes from pl to bp
    - Reads other user’s data, including secret keys, if payload value is too big



How Heartbleed Works: The Code Behind the Internet's Security Nightmare

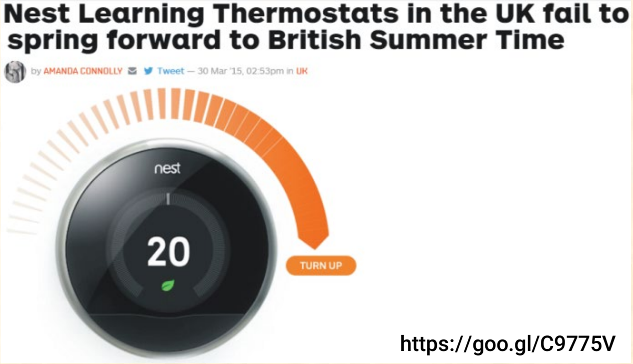
Eric Limer

Filed to: HEARTBLEED 4/09/14 2:59pm

https://goo.gl/1Joxy2

By now you've surely heard of Heartbleed, the hole in the internet's security that exposed countless encrypted transactions to any attacker who knew how to abuse it. But how did it actually work? Once you break it down, it's actually incredibly simple. And a little hilarious. But mostly terrifying.

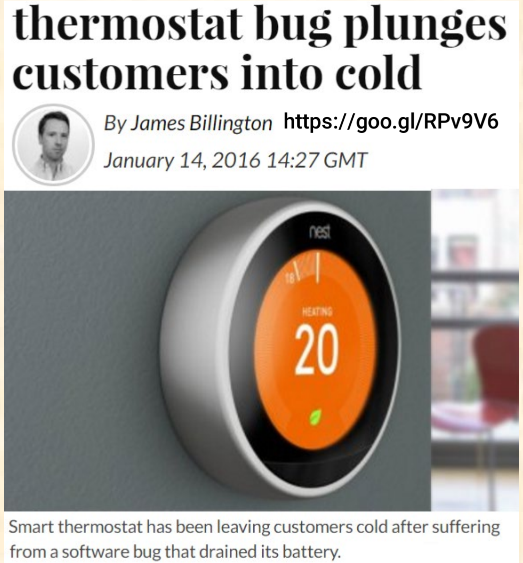
Large Scale Production = Big Problems



Nest Learning Thermostats in the UK fail to spring forward to British Summer Time

by AMANDA CONNOLLY ✉ Tweet — 30 Mar '15, 02:53pm in UK

https://goo.gl/C9775V

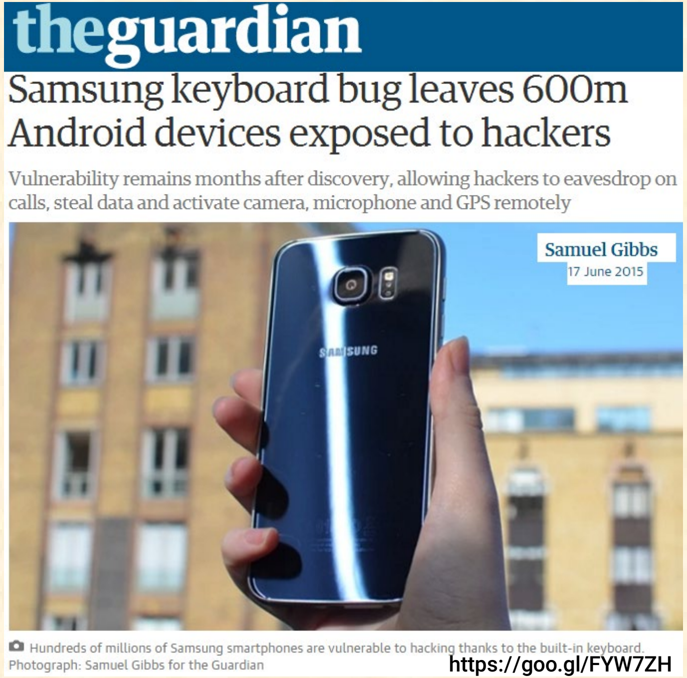


thermostat bug plunges customers into cold

By James Billington https://goo.gl/RPv9V6

January 14, 2016 14:27 GMT

Smart thermostat has been leaving customers cold after suffering from a software bug that drained its battery.



theguardian

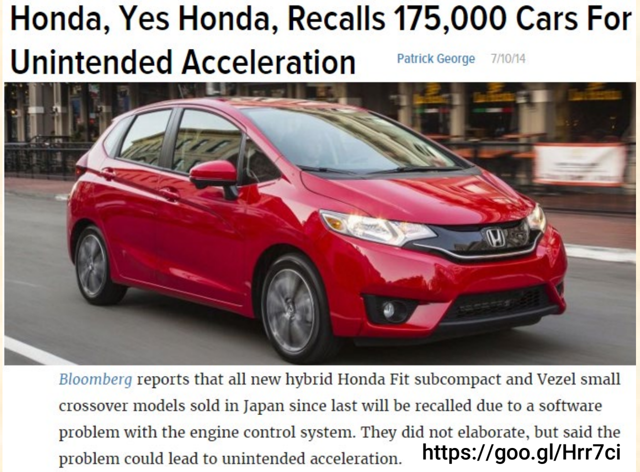
Samsung keyboard bug leaves 600m Android devices exposed to hackers

Vulnerability remains months after discovery, allowing hackers to eavesdrop on calls, steal data and activate camera, microphone and GPS remotely

Samuel Gibbs
17 June 2015

Hundreds of millions of Samsung smartphones are vulnerable to hacking thanks to the built-in keyboard. Photograph: Samuel Gibbs for the Guardian

https://goo.gl/FYW7ZH



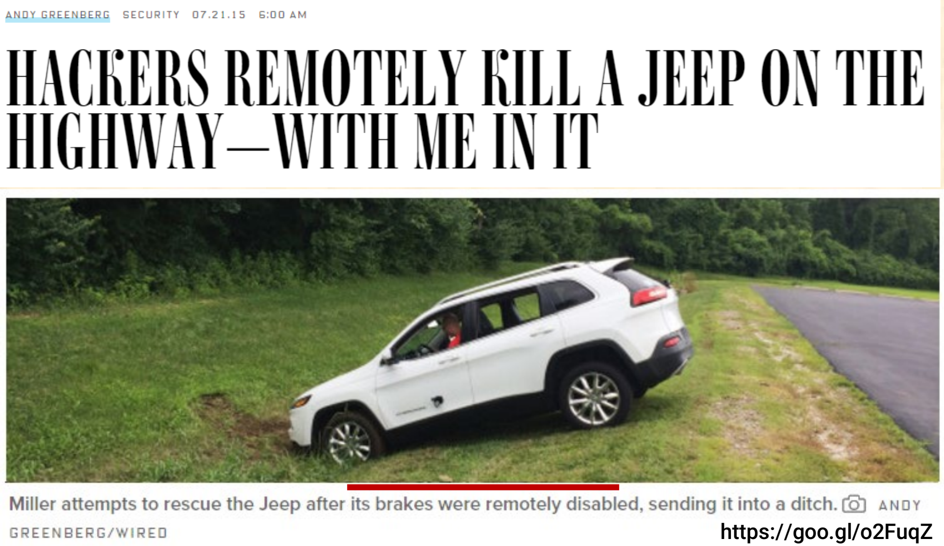
Honda, Yes Honda, Recalls 175,000 Cars For Unintended Acceleration

Patrick George 7/10/14

Bloomberg reports that all new hybrid Honda Fit subcompact and Vezel small crossover models sold in Japan since last will be recalled due to a software problem with the engine control system. They did not elaborate, but said the problem could lead to unintended acceleration. https://goo.gl/Hrr7ci

There Are Too Many Examples

- A steady stream of software mishaps, recalls, etc.

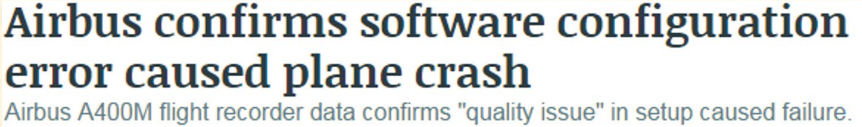


ANDY GREENBERG SECURITY 07.21.15 6:00 AM

HACKERS REMOTELY KILL A JEEP ON THE HIGHWAY—WITH ME IN IT

Miller attempts to rescue the Jeep after its brakes were remotely disabled, sending it into a ditch. ANDY GREENBERG/WIRED

https://goo.gl/o2FuqZ



Airbus confirms software configuration error caused plane crash

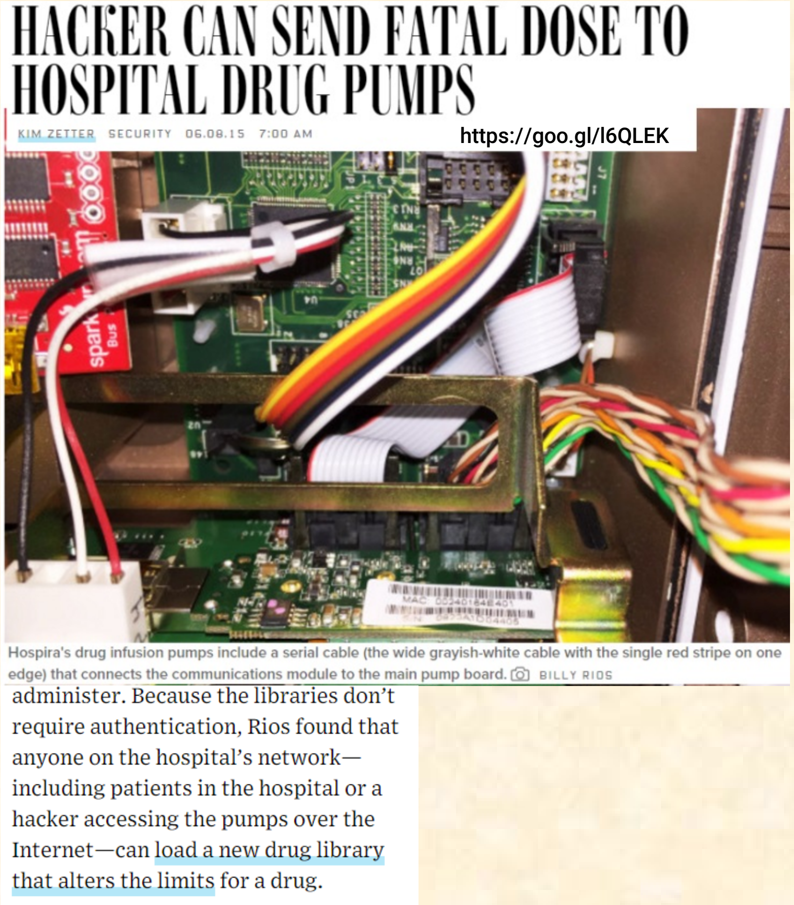
Airbus A400M flight recorder data confirms "quality issue" in setup caused failure.

https://goo.gl/4hbom9

https://goo.gl/l2RWUv

This Goes Far Beyond Transportation

Carnegie Mellon University



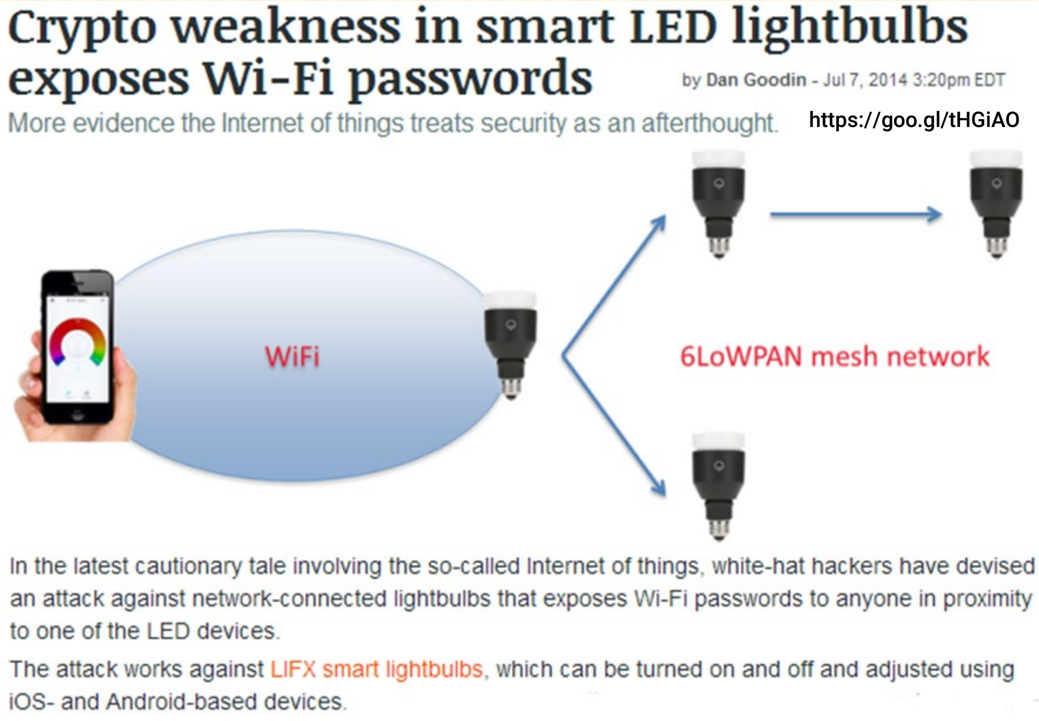
HACKER CAN SEND FATAL DOSE TO HOSPITAL DRUG PUMPS

KIM ZETTER SECURITY 06.08.15 7:00 AM

https://goo.gl/I6QLEK

Hospira's drug infusion pumps include a serial cable (the wide grayish-white cable with the single red stripe on one edge) that connects the communications module to the main pump board. BILLY RIOS

administer. Because the libraries don't require authentication, Rios found that anyone on the hospital's network—including patients in the hospital or a hacker accessing the pumps over the Internet—can load a new drug library that alters the limits for a drug.



Crypto weakness in smart LED lightbulbs exposes Wi-Fi passwords

by Dan Goodin - Jul 7, 2014 3:20pm EDT

More evidence the Internet of things treats security as an afterthought.

https://goo.gl/tHGiAO

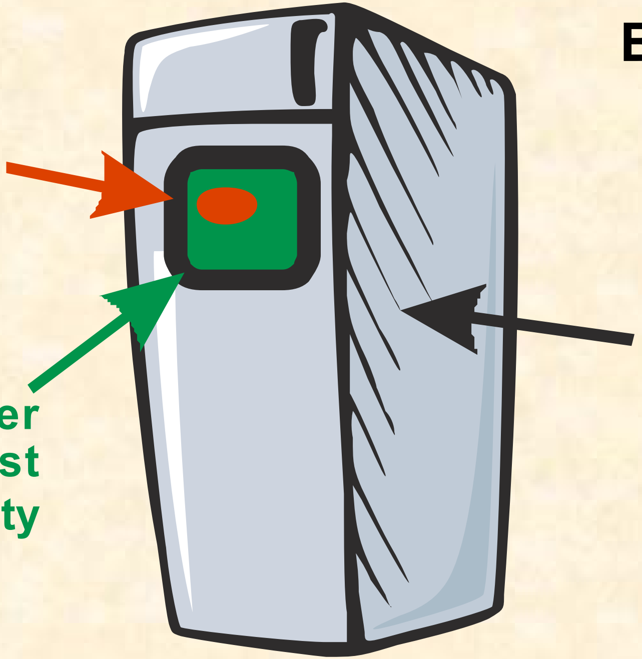
In the latest cautionary tale involving the so-called Internet of things, white-hat hackers have devised an attack against network-connected lightbulbs that exposes Wi-Fi passwords to anyone in proximity to one of the LED devices.

The attack works against LIFX smart lightbulbs, which can be turned on and off and adjusted using iOS- and Android-based devices.

© 2021 Philip Koopman

Act As If Your Products Live Or Die By Their Software

BOM = Bill Of Materials



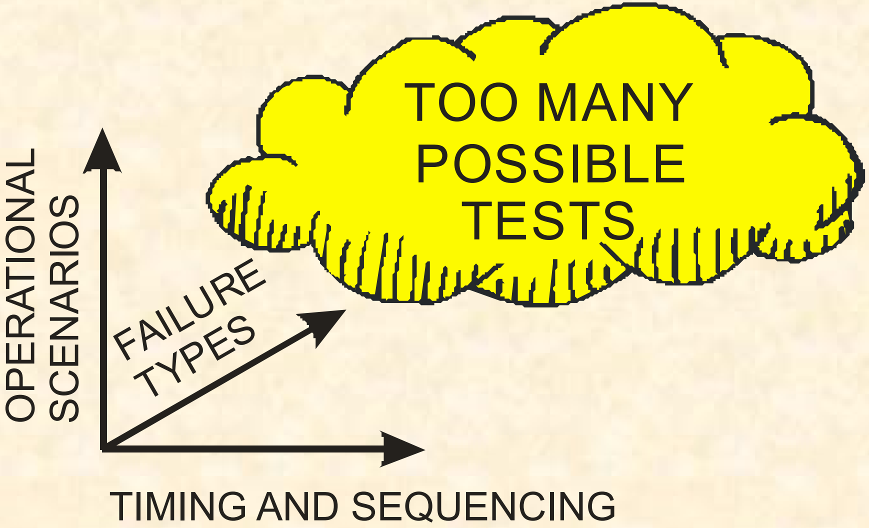
Software
0% of BOM cost
90% of product differentiation

Electronic Controller
10% of BOM cost
Mostly commodity

Mechanical System
90% of BOM cost
Mostly commodity

Product Testing Won't Find All Bugs

- Testing bad software simply makes it less bad
  - Testing cannot produce good software all on its own



- One third of faults take more than 5000 years to manifest

Adams, N.E., "Optimizing preventive service of software product," IBM Journal of Research and Development, 28(1), p. 2-14, 1984. (Table 2, pg. 9, 60 kmonth column)

  - Your customers will regularly experience bugs that you will not see during testing
  - For most products, you can't even test 5000 years

How Bad Can It Possibly Be?

- For YOUR product, what is the worst possible outcome:
  - For a software bug?
    - People killed or injured?
    - Property damage?
    - Cost to deploy a fix?
    - Loss of brand reputation?
  - For a malicious attack?
  - Hint: *The answer is the same* for both bugs and successful attacks

https://goo.gl/IUFUPG

https://goo.gl/OSfG8i

- Regulation is likely to increase
  - IEC 60730 safety standard required for European appliances
  - Security standards are already proliferating

Designing For Safety

- Every system is assumed to be unsafe by default
  - It is up to you to proactively show that it is safe
    - » Example: DEF STAN 00-55 Parts 1 & 2

1. Collect risks
   - What can go wrong? What does "safe" really mean?
2. Assign risk severity
   - What types of mishaps are most important to avoid?
3. Perform risk mitigation
   - How can you avoid hazards and activation of hazards?
4. Develop software to acceptable level of integrity
   - Ensure that risk mitigation is successful

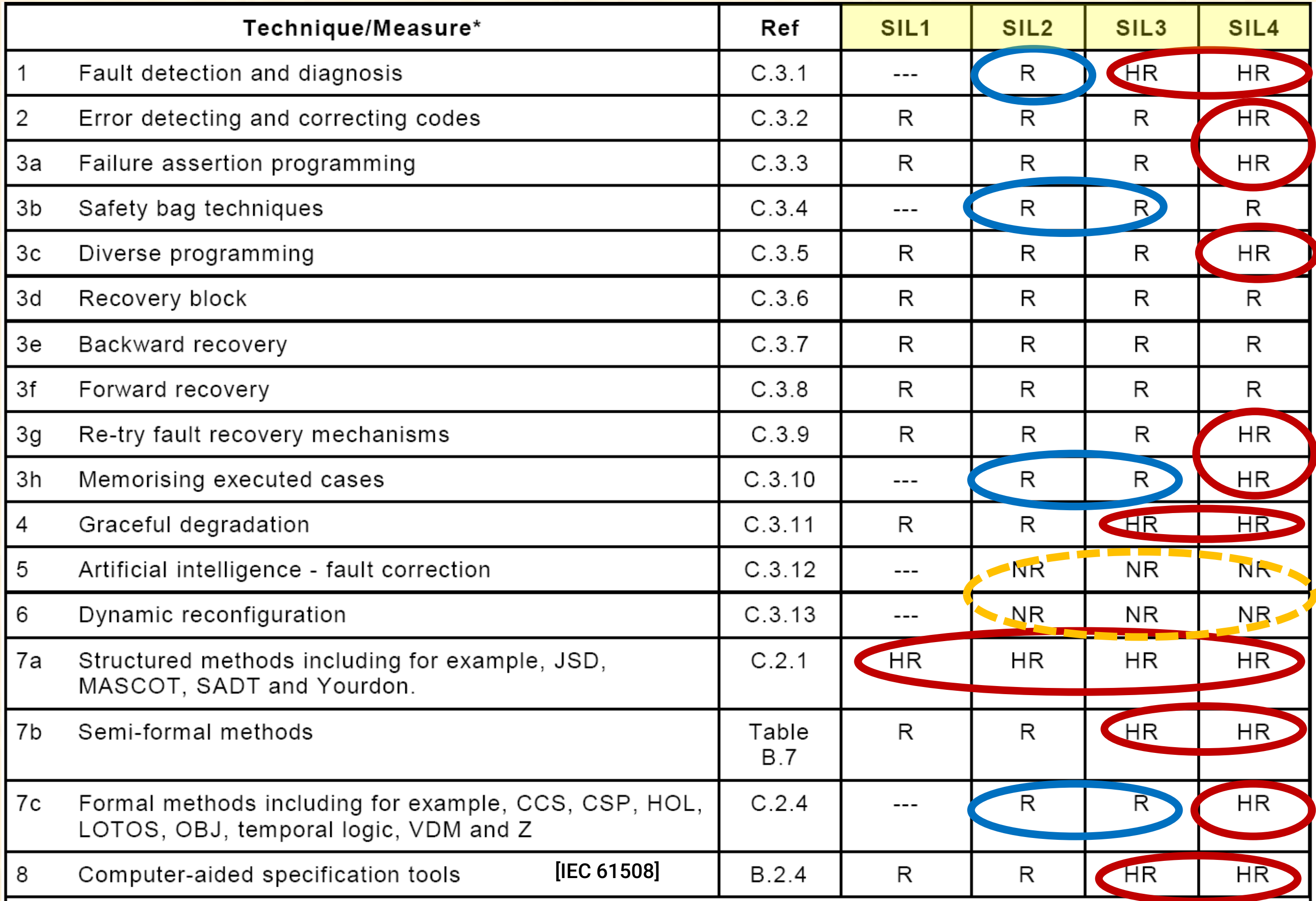
Risk Identification & Assessment

- Create a **Hazard Log** (list of hazards), including HAZOP
- **PHA** (Preliminary Hazard Analysis) & **Risk Table**
  - E.g. Consequence
    - $100M loss
    - $1M loss
    - …
    - $100 loss
  - E.g. Probability
    - Every minute
    - Weekly
    - …
    - Every 10 years
  - (4) .. (0) ➔ See SIL on next slide

EXAMPLE RISK		Probability				
		Very High	High	Medium	Low	Very Low
Conse-quence	Very High	Very High (4)	Very High (4)	Very High (4)	High (3)	High (3)
	High	Very High (4)	High (3)	High (3)	Medium (2)	Medium (2)
	Medium	High (3)	High (3)	Med. (2)	Med. (2)	Low (1)
	Low	High (3)	Medium (2)	Medium (2)	Low (1)	Very Low (0)
	Very Low	Medium (2)	Low (1)	Low (1)	Very Low (0)	Very Low (0)

Higher SIL Invokes Engineering Rigor

- SIL = Safety Integrity Level
  - SIL4 = catastrophic
  - SIL1 = minor injuries
  - Used to determine required level of engineering rigor
- Example: IEC 61508
  - HR= Highly Recommended
  - R = Recommended
  - NR = Not Recommended (don't do this)



	Technique/Measure*	Ref	SIL1	SIL2	SIL3	SIL4
1	Fault detection and diagnosis	C.3.1	---	R	HR	HR
2	Error detecting and correcting codes	C.3.2	R	R	R	HR
3a	Failure assertion programming	C.3.3	R	R	R	HR
3b	Safety bag techniques	C.3.4	---	R	R	R
3c	Diverse programming	C.3.5	R	R	R	HR
3d	Recovery block	C.3.6	R	R	R	R
3e	Backward recovery	C.3.7	R	R	R	R
3f	Forward recovery	C.3.8	R	R	R	R
3g	Re-try fault recovery mechanisms	C.3.9	R	R	R	HR
3h	Memorising executed cases	C.3.10	---	R	R	HR
4	Graceful degradation	C.3.11	R	R	HR	HR
5	Artificial intelligence - fault correction	C.3.12	---	NR	NR	NR
6	Dynamic reconfiguration	C.3.13	---	NR	NR	NR
7a	Structured methods including for example, JSD, MASCOT, SADT and Yourdon.	C.2.1	HR	HR	HR	HR
7b	Semi-formal methods	Table B.7	R	R	HR	HR
7c	Formal methods including for example, CCS, CSP, HOL, LOTOS, OBJ, temporal logic, VDM and Z	C.2.4	---	R	R	HR
8	Computer-aided specification tools	B.2.4	R	R	HR	HR

[IEC 61508]

Head Count: Half Designers, Half Testers

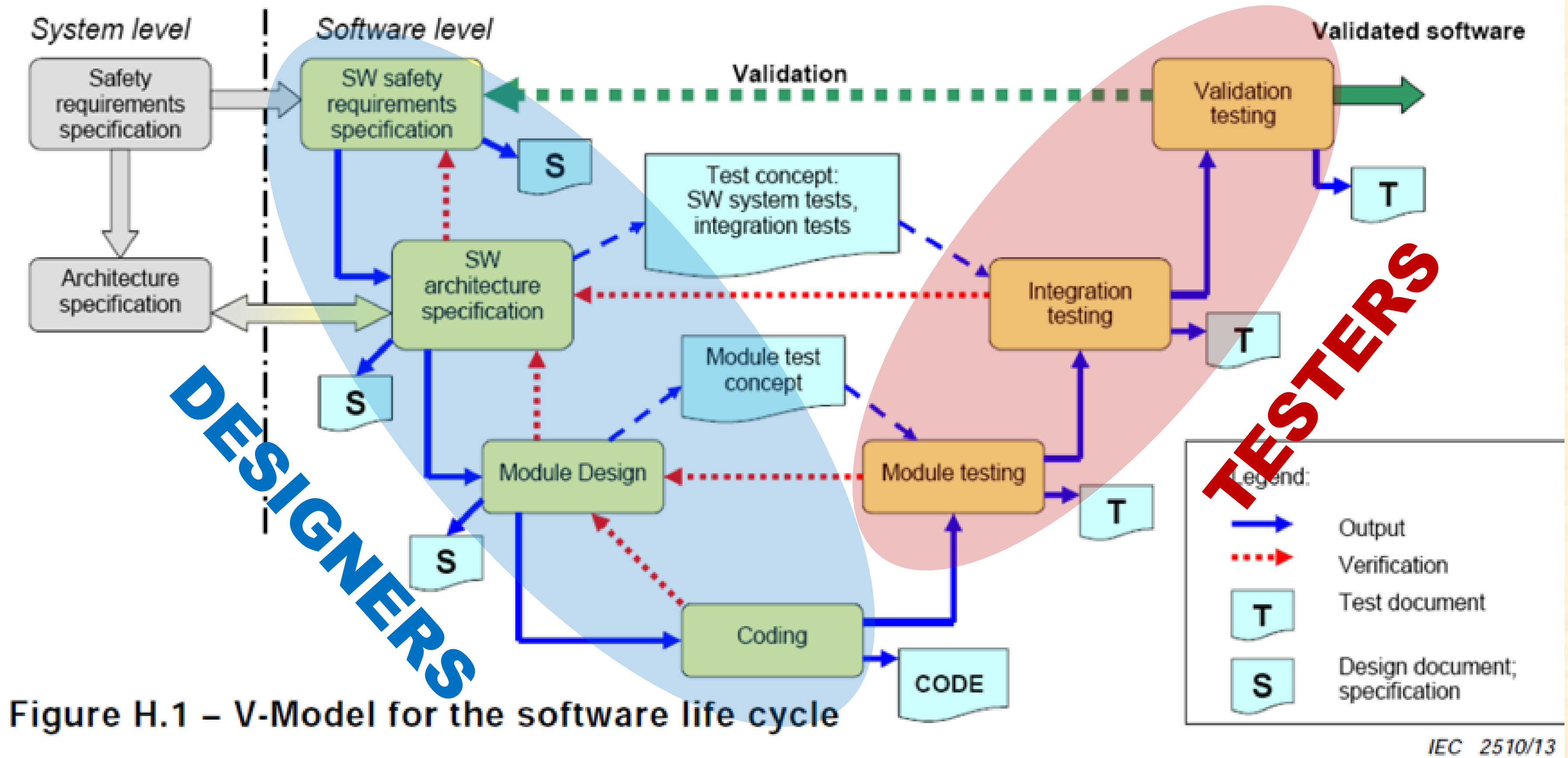


Figure H.1 – V-Model for the software life cycle

IEC 60730 Appliance Safety

[IEC 60730]

Essential Practice: Peer Reviews

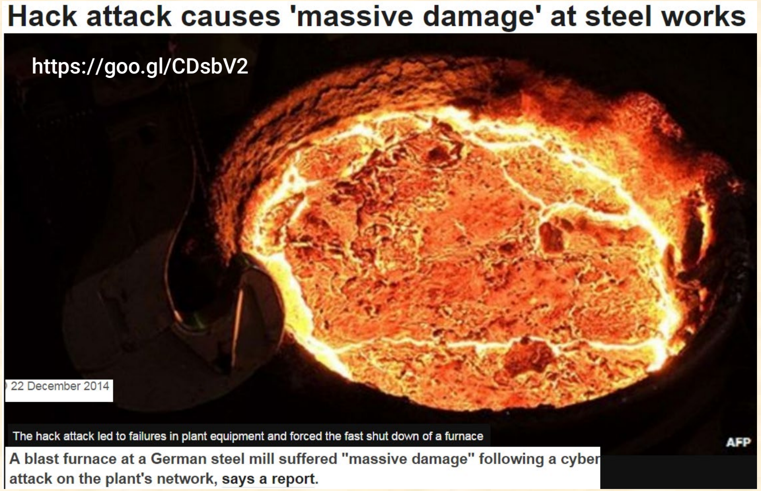
Carnegie Mellon University

- Gold Standard: Fagan Style Inspection
  - Pre-review meeting
  - Formal meeting
  - Written review report
  - Follow-up and possible re-inspection
  - The more formal the review, the higher the payoff

- Good reviews find 50%+ of defects for about 10% of project cost
  - Defects are found early, when they are cheaper to fix and cause less disruption
  - *Why is it so many designers say they don’t have time to do peer reviews?*

- Other technical issues are crucial for good embedded software
  - Watchdog timers, mutexes, Rate Monotonic Scheduling, interrupts, exception handling, reducing code complexity, secure update, timekeeping, performance optimization, …

Security Matters for Industrial Systems!



Hack attack causes 'massive damage' at steel works

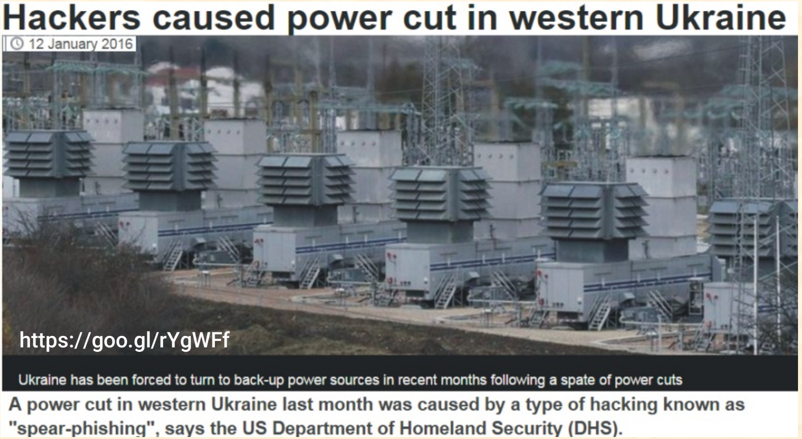
https://goo.gl/CDsbV2

22 December 2014

The hack attack led to failures in plant equipment and forced the fast shut down of a furnace

AFP

A blast furnace at a German steel mill suffered "massive damage" following a cyber attack on the plant's network, says a report.



Hackers caused power cut in western Ukraine

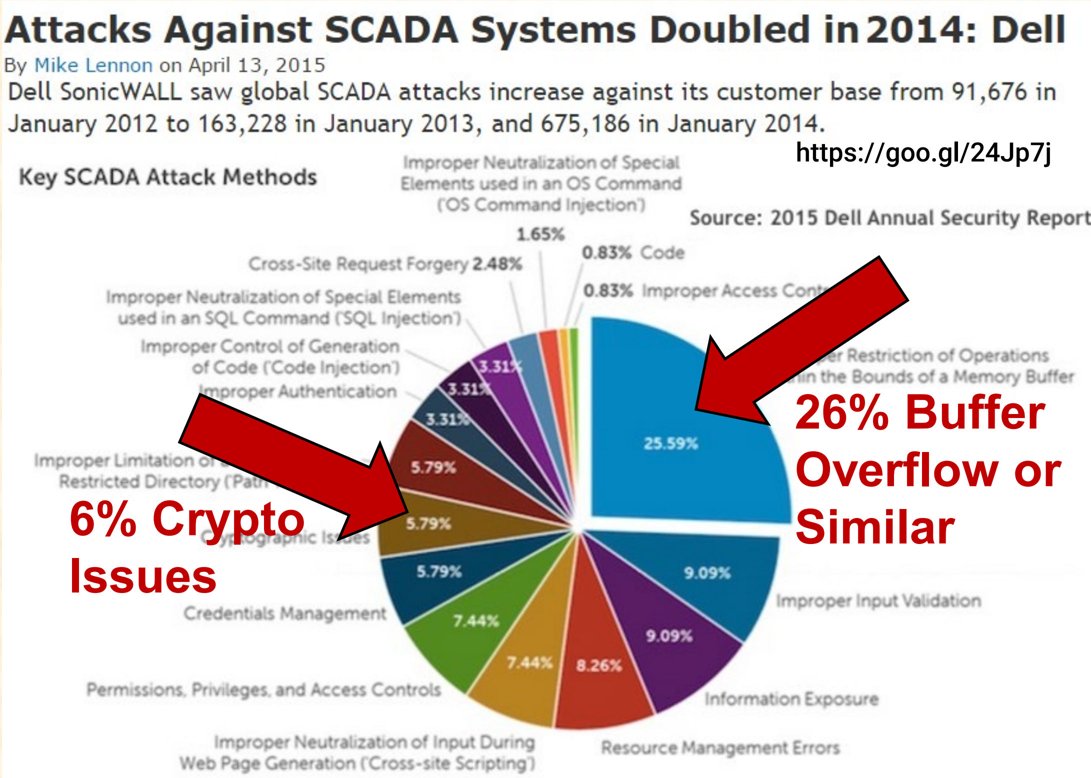
12 January 2016

https://goo.gl/rYgWFf

Ukraine has been forced to turn to back-up power sources in recent months following a spate of power cuts

A power cut in western Ukraine last month was caused by a type of hacking known as "spear-phishing", says the US Department of Homeland Security (DHS).

- Attacks can affect the physical world



Attacks Against SCADA Systems Doubled in 2014: Dell

By Mike Lennon on April 13, 2015

Dell SonicWALL saw global SCADA attacks increase against its customer base from 91,676 in January 2012 to 163,228 in January 2013, and 675,186 in January 2014.

https://goo.gl/24Jp7j

Key SCADA Attack Methods

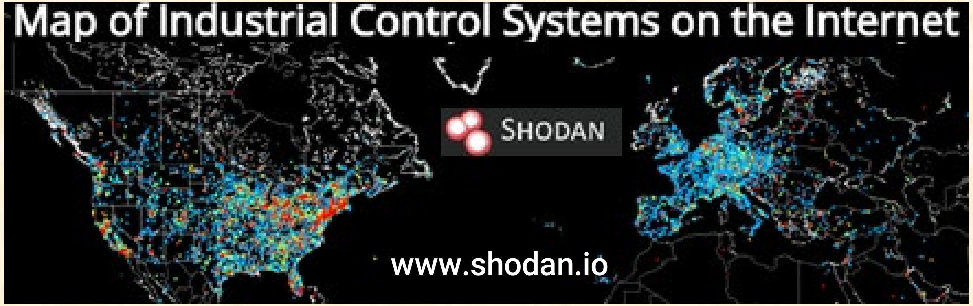
Source: 2015 Dell Annual Security Report

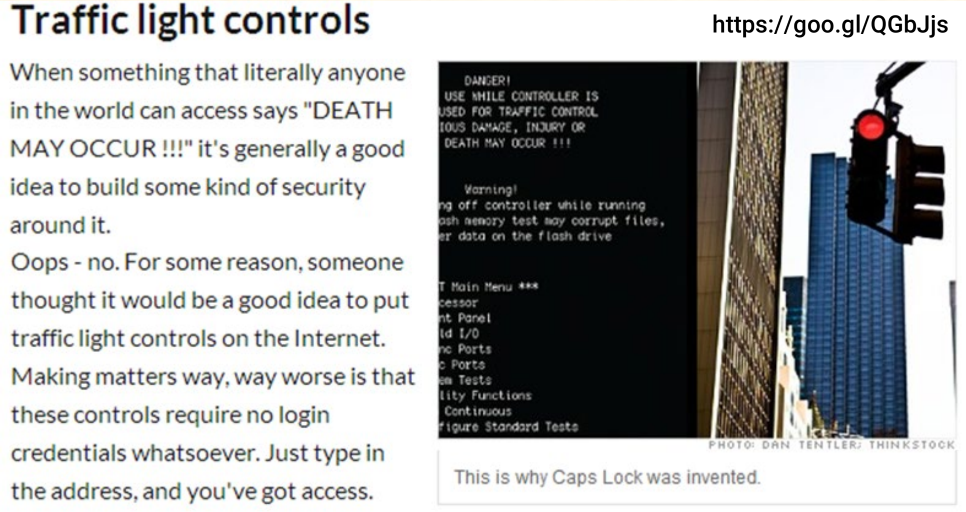
26% Buffer Overflow or Similar

6% Crypto Issues

Industrial Controls Are Targets

- The Bad Guys are after more than credit card numbers





Traffic light controls

https://goo.gl/QGbJjs

When something that literally anyone in the world can access says "DEATH MAY OCCUR !!!" it's generally a good idea to build some kind of security around it.

Oops - no. For some reason, someone thought it would be a good idea to put traffic light controls on the Internet. Making matters way, way worse is that these controls require no login credentials whatsoever. Just type in the address, and you've got access.

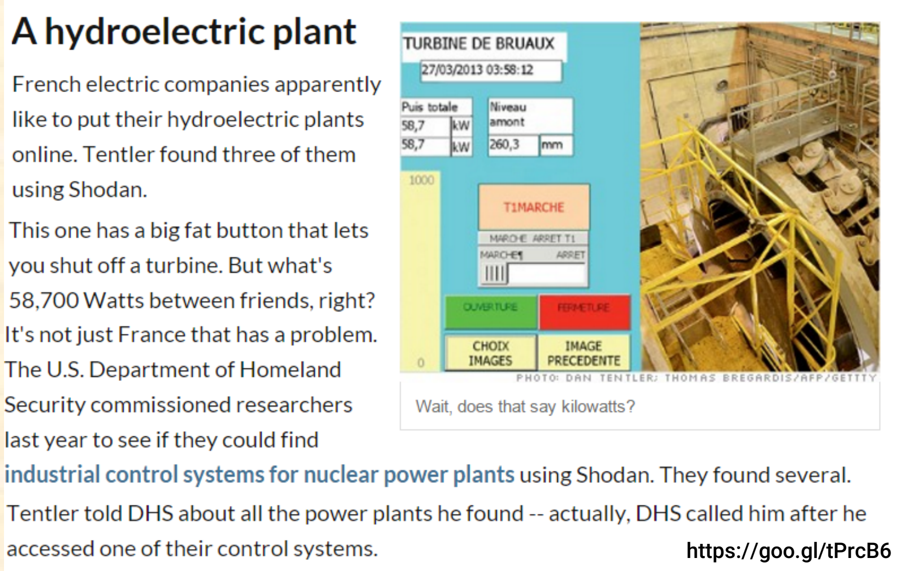
```
    DANGER!
 USE WHILE CONTROLLER IS
USED FOR TRAFFIC CONTROL
IOUS DAMAGE, INJURY OR
 DEATH MAY OCCUR !!!


    Warning!
ng off controller while running
ash memory test may corrupt files,
er data on the flash drive


T Main Menu ***
cessor
nt Panel
ld I/O
nc Ports
c Ports
em Tests
lity Functions
 Continuous
figure Standard Tests
```

PHOTO: DAN TENTLER; THINKSTOCK

This is why Caps Lock was invented.



A hydroelectric plant

French electric companies apparently like to put their hydroelectric plants online. Tentler found three of them using Shodan.

This one has a big fat button that lets you shut off a turbine. But what's 58,700 Watts between friends, right?

It's not just France that has a problem. The U.S. Department of Homeland Security commissioned researchers last year to see if they could find industrial control systems for nuclear power plants using Shodan. They found several.

Tentler told DHS about all the power plants he found -- actually, DHS called him after he accessed one of their control systems.

PHOTO: DAN TENTLER; THOMAS BREGARDIS/AFP/GETTY

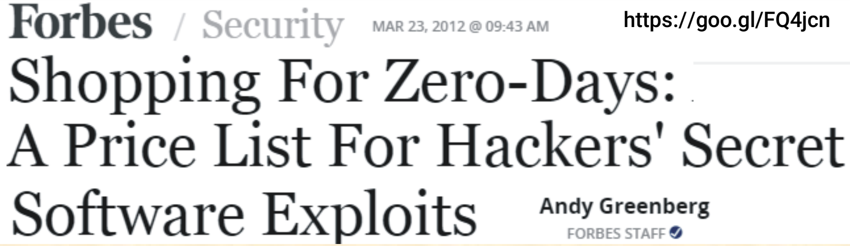
Wait, does that say kilowatts?

https://goo.gl/tPrcB6

"a big fat button lets you shut off a turbine"
(No login credentials required)

Designing For Security

- Security testing isn't enough
  - Bad code is especially vulnerable
  - Testing mostly finds <u>known</u> problems
- Need to address:
  - Security requirements
  - Characterize threats & risks
  - Security risk management plan
  - Deploying security patches
- Myriad technical issues
  - Secure update, cryptography, input validation, least privilege, code quality, passwords, privacy, web interface, error handling, secure coding, …

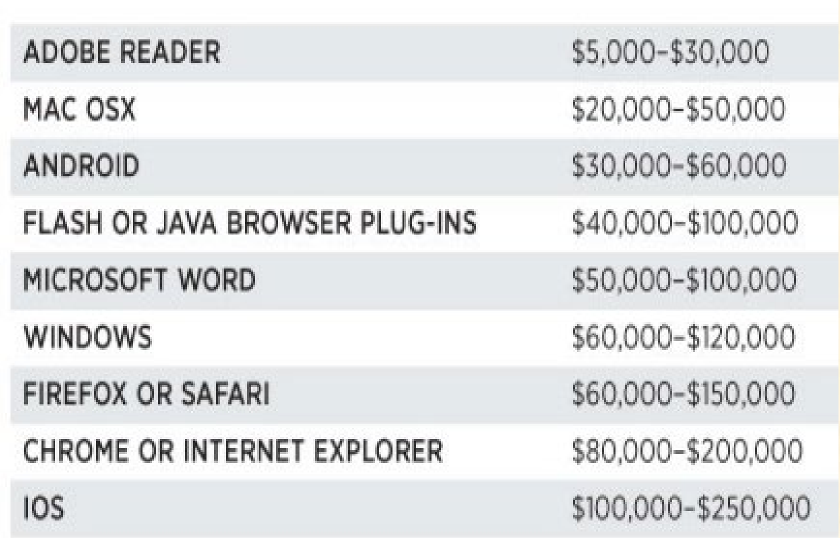


Forbes / Security MAR 23, 2012 @ 09:43 AM

https://goo.gl/FQ4jcn

Shopping For Zero-Days: A Price List For Hackers' Secret Software Exploits

Andy Greenberg
FORBES STAFF



ADOBE READER	$5,000-$30,000
MAC OSX	$20,000-$50,000
ANDROID	$30,000-$60,000
FLASH OR JAVA BROWSER PLUG-INS	$40,000-$100,000
MICROSOFT WORD	$50,000-$100,000
WINDOWS	$60,000-$120,000
FIREFOX OR SAFARI	$60,000-$150,000
CHROME OR INTERNET EXPLORER	$80,000-$200,000
IOS	$100,000-$250,000

Testing Alone Won't Fix Bad Software

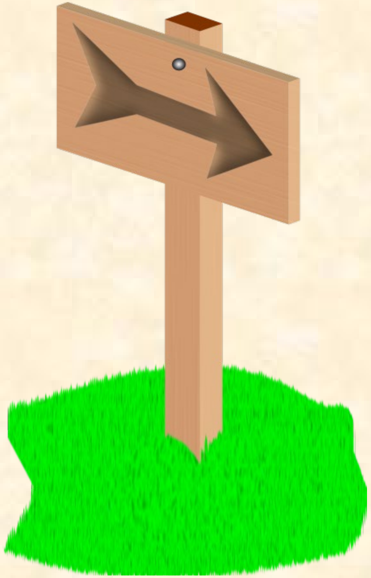
- You can't test in quality, safety, or security
- In an ideal world,
  throw it away and start over
  - But, the world is not ideal …
- Incremental Reengineering
  - Identify & fix high risk modules
  - Clean sheet for each module; don't try to derive design from code
- Improvement requires cultural change
  - Requires commitment to good software at all levels of organization
  - Commitment must survive a "but we have to ship next week" crisis

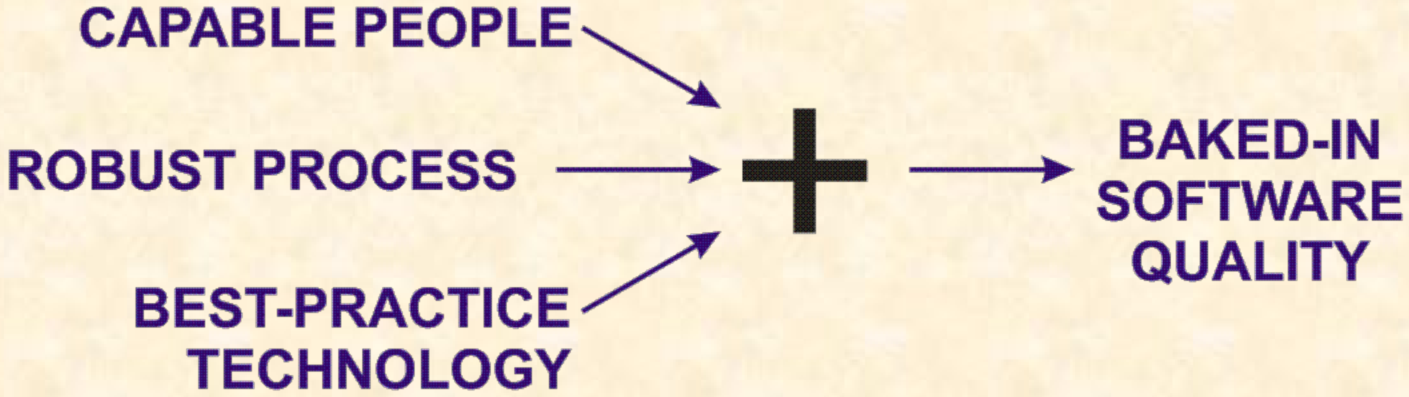
https://goo.gl/i7Ue6N

Top 10 Embedded SW Warning Signs

1. Software time estimates are driven by external dates
2. Process steps skipped during schedule crunches
3. Software development is simply “coding” plus “testing”
4. Poor traceability from product test to requirements
5. Bugs due to poor code style & complexity
6. Bugs in software fault detection/recovery
7. No Security Plan; no Safety Plan
8. Tester:Developer ratio too far from about 1 : 1
9. More than about 5-10% of bugs are found in product test
10. Fewer than 50% of defects are found by peer review

The Path To Good Software





CAPABLE PEOPLE

ROBUST PROCESS

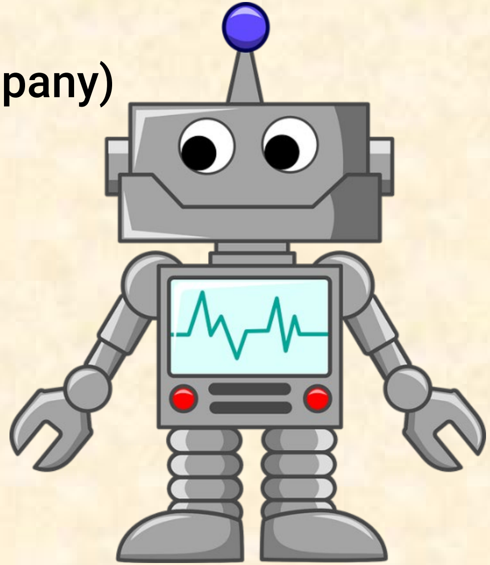
BEST-PRACTICE TECHNOLOGY

+

BAKED-IN SOFTWARE QUALITY

Software Quality, Safety & Security

- Software is crucial for providing value
  - But – even a single line of bad code can kill a product (or a company)
  - Writing software is a high-stakes profession. Take it seriously.
- Good software requires process + technology + people
  - Embedded software requires unique technical approaches
  - You can't test quality, safety, or security into software
- Good process enables good software
  - Whether "V" or agile, need to actually follow a good process
  - Typically need 1:1 head count for testers:developers
  - Peer reviews find 50%+ of defects on the cheap – why aren't you doing them?
- Safety and security are essential – don't wait until there is a loss event
  - Most embedded software is safety critical or mission critical
  - Security is required in essentially all embedded software



What Happens Next?

- Assess where you are
  - How good is your code quality?
  - How good are your software, process & technical skills?
  - How good are your safety & security practices?
- Improve process, skills, technology
  - Ensure you are doing effective peer reviews
  - Formalize and follow a reasonable software process
  - Adopt/adapt relevant safety & security standards
  - Ensure developers have strong embedded software & process skills
- Cultural change
  - Make software quality a first class company goal, not a sideline
  - Daily practices, process support, training, metrics