

Regulatory Tales: Sparks, Carrots, and Sticks

Cybersecurity Seminar Series
UCSF-Stanford CERSI-FDA Distinguished Speaker Series on Cybersecurity
for Biomedical Engineering

Nov. 18, 2021

Carl Landwehr

Visiting Professor

*Dept. of Electrical and Computer Engineering
University of Michigan*

The history of attempts to regulate technology is long and varied. This talk will reviews examples from a range of technical areas, some successful, some less so, and will draw a few lessons from them. Topic areas will include building construction, automobiles, airplanes, cybersecurity, and perhaps others. Discussion will be encouraged.

The talk on one slide



- Technology has been regulated for a long time
- Regulations typically involve a spark, a carrot, and a stick
- How these elements are crafted can make a big difference in the effectiveness of a regulation
- The evolution of the software and digital systems marketplace has not favored investment in high-assurance software development, except where the public has demanded it
- Developing the incentives for high assurance development for critical devices and systems should be a focus for both regulators and software professionals

I've been advised that it's a good thing to be able to put the entire talk onto one slide, just in case anyone loses attention or interest, and this is my attempt to do that.

What I want to talk about today is how regulation of technology has developed in several different domains. I expect many of you in the audience are involved in designing or certifying systems that may be life-critical in the biomedical domain. You probably all know much more than I do about your specific domains of expertise. So I thought it might be interesting to spend a little time raising our eyes to some different domains in which lives are at stake and to see how regulation has developed over time in them. Each domain has different characteristics – different industries are involved, public interests in the results may differ, politics often plays a role. Reviewing these examples may help us understand how regulation in new fields like software development might be structured and improved.

The earliest regulation?

229. If a builder build a house for some one, and does not construct it properly, and the house which he built fall in and kill its owner, then that builder shall be put to death.¹

About 1772 BCE

1. as translated by L. W. King, available at:
<https://avalon.law.yale.edu/ancient/hamframe.asp>



Image: Wikipedia, public domain

One of the earliest forms of regulation I'm aware of is a piece of Hammurabi's code. The entire code includes 282 items that range from criminal activity to ferry tolls and the price of houses. Numbers 228 – 233 concern the construction of houses. Section 229 shown here is perhaps the most dramatic, calling for the death of the builder in case the house he built falls in and kills the owner. If we think of this as a building code, it would be in the category of a "performance code" today: it doesn't tell you how to build the building, but if it doesn't stand up, you are liable for the consequences. So in effect, the builder warrants the building with his life. We must imagine that these laws didn't arise without some stimulus – houses must on occasion have fallen in, sparking the development of this code, which is surely in the nature of a "stick."



Let's look at a few more historical examples that sparked development of the building codes we have today. In most cases I looked at, the sparking events were disasters: fires, earthquakes, storms, and so on.

1. The Great Fire of London in September 2-5 1666 destroyed 430 acres, about 80% of the city at that time. Documented by Samuel Pepys in his famous diary, it led to the 1667 London Rebuilding Act which called for new construction to be faced with brick and imposed other measures designed to reduce the likelihood of large fires.

2. Jumping ahead to the 20th century, consider earthquakes in California. The most famous of course is the San Francisco earthquake of 1906. A month after the event, scientists and engineers banded together to form the Structural Association of San Francisco and concluded that well-braced wooden buildings, secured strongly to their foundations could have withstood that quake. However, the city fathers declined to add earthquake resistance to the building code and, perhaps in order not to scare off developers, blamed much of the devastation on fires. Ordinances were passed approving the use of reinforced concrete and requiring steel framing in any new brick construction. Not until 1925, when Santa Barbara suffered a severe earthquake that leveled most of its downtown, did requirements that structures be designed to withstand horizontal forces – first seismic code requirement – enter the building codes in California. (MCEER website)

3. Miami was hit by a powerful hurricane in 1926, and another category 4 hurricane











struck further north in 1928 in the Lake Okeechobee area, causing thousands of deaths and severe property damage. Buildings made of brick or stone survived better than others, and this observation led to stronger building codes. As in California, only after the second disaster were the codes modified. Note however that deficiencies in code enforcement led to severe damage from Hurricane Andrew in 1992.

4. Collapse of Hyatt Regency Hotel Skywalk in Kansas City, July, 1981. A suspended walkway in the atrium of a large hotel collapsed under the burden of a large crowd at a Friday afternoon tea dance, killing 114 people and injuring more than 200 others. The cause was ultimately determined to be a change from the original design in which the walkways were suspended. The change had been approved; the fundamental problem seems to have been lack of proper communication between the designer and the steel company. Responsible engineers were convicted of gross negligence and unprofessional conduct; they lost their licenses and ASCE memberships, as did the firm.

5. Oklahoma City bombing of the Murrah Building, April 1995. This malicious attack by a domestic terrorist claimed 168 lives and damaged 324 buildings. The effect of this was to trigger the construction of Jersey walls around many existing Federal buildings and to add new setback requirements for new federal construction.

6. World Trade Center collapse, 2001: Again, a malicious attack. In 2004 New York adopts Local Law 26 with code revisions; Sept. 2008, Intl. Code Council adopts 23 changes to fire and building codes motivated by lessons from WTC collapse

Building Code Characteristics

-  • Can specify performance (withstand wind of 100 MPH) or construction (brick or stone facing)
-  • Design approval (building permit)
-  • Inspection during construction
-  • Approval before occupancy
-  • Buyer / institutional confidence in the project.
-  • Standard international codes can be tailored for local risk environments
-  • Standardization of requirements simplifies design and construction
-  • Enforcement is local
-  • Code can evolve to keep up with new technologies and risks
-  • Material suppliers, architects, construction firms collaborate on updates; updates stimulate market for products

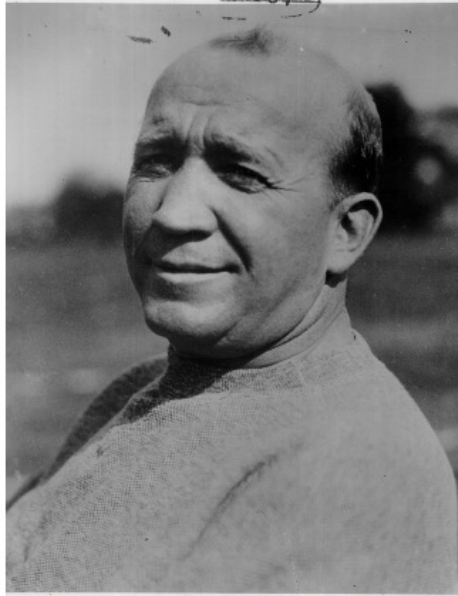
Let's review a few features of building codes and consider the carrots and sticks they represent.

Note that a code can specify construction details, such as what kinds of materials are allowed in the outside of buildings or alternatively can specify that the building withstand a wind of a certain velocity without specifying how that is to be achieved. From the standpoint of sparking innovation, the performance specification seems better, but it comes with a requirement for demonstrating, somehow, that the specification is met.

Note that inspections occur as the building is constructed -- when the footings are poured, when the plumbing and electrical systems are completed, and so on .

Designs must be approved as conforming to code prior to the start of construction, and an occupancy permit, certifying that the completed structure conforms to the code, must be obtained before the owner can move in. These are sticks.

But there are some carrots as well. First, the buyer (and the financial institution that may have lent the money for the project) gains improved confidence in the product. Codes can be tailored to reflect local risks (earthquakes vs hurricanes for example), and they are enforced locally. They can evolve as technologies and conditions change, and that evolution can help build markets for new products.



What does football
have to do with
aviation regulation?

Famed Notre Dame football
coach Knute Rockne,
killed in the crash of a
Fokker F-10 trimotor near
Wichita in 1931

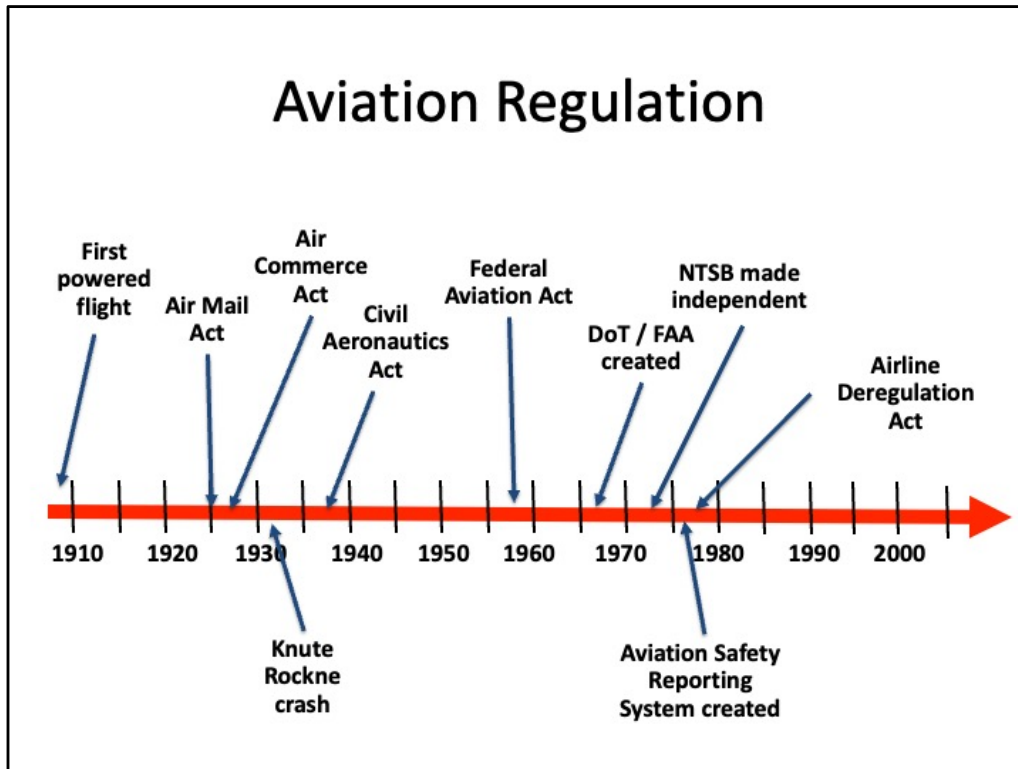
<https://chicago.suntimes.com/2021/3/29/22356683/knute-rockne-plane-crash-death-faa-federal-aviation-administration-airplane-safety-jim-lefebvre>



You may not recognize this man, but he is Knute Rockne, the nationally famous football coach at Notre Dame in the 1920s. He traveled around the country giving motivational talks just as coaches do today. In late March, 1931, he was en route to California. He took a train to Kansas City and planned to continue from there by air to Los Angeles. His flight was on a Fokker F10A trimotor, which sported plywood wings. The flight encountered bad weather near Wichita but decided to proceed. The flight subsequently crashed, killing all 8 people on board, with icing and failure of one of the wings eventually blamed for the disaster. There was substantial outcry from the public, and all F10s and F10As were grounded.

The history of aviation regulation is complex and well-documented on the FAA's website and elsewhere.

But there is little doubt that the aircraft industry itself played a significant role in creating and working with its own regulatory infrastructure. The industry realized that it could only prosper if air travel was seen as safe, and that the government had a key role to play in overseeing the safety of aircraft, pilots, and the air transport system.



This slide highlights just a few of the many events in the history of the development and regulation of aviation.

The development of commercial aviation was intentionally stimulated by the Post Office department, first with contracts to deliver mail, and then with the requirement that mail contractors also have the ability to take passengers on board their aircraft. The industry recognized from the beginning the need to establish public trust in the safety of aviation, and so it supported the idea that the Commerce Department would develop and enforce air traffic rules, license pilots, establish airways, and operate navigational aids in the Air Commerce Act of 1926.

But continuing crashes eventually sparked the 1938 Civil Aeronautics Act, which established the Civil Aeronautics Authority and its Air Safety Board to investigate crashes and recommend ways to prevent accidents. Eventually it was recognized that accident investigation needed to be independent, and in 1966 the National Transportation Safety Board was created as a separate body within DoT to investigate accidents and recommend remedies.

In 1974, Congress moved NTSB outside of DoT to assure the independence of its investigations. Importantly, the NTSB was NOT given the authority to regulate, so its recommendations had to be adopted by the FAA before they would have any force. The Airline Deregulation Act of 1978 was strongly influenced by the opinion at the time that government needed to regulate less, not more. But it deregulated the setting of fares only, not the safety regulation.

FAA Memo on Fly-by-Wire Flight Control System – 1974

DEPARTMENT OF TRANSPORTATION
FEDERAL AVIATION ADMINISTRATION

WASH-1000, D.C. 20591

MAY 2 1974

Dr. Peter R. Kurzahls, Director
Guidance, Control & Information Systems, Code RE
NASA Headquarters
Washington, D.C. 20546

Dear Dr. Kurzahls:

This is in response to the inquiry from your office regarding quantification of probability terms used in the context of reliability of software systems.

Section 13.1008 of the Federal Aviation Regulations is designed to ensure that the flight characteristics of aircraft in and around which would present the most significant safety hazard are not compromised by unreliability of the flight control system. It has required substantiating data for which even by criteria that the aircraft must achieve a reliability of 10⁻⁹.

To date, this criteria has been met by the use of redundant systems, built-in tests, and automatic reversion to manual control for the system. It is not possible to first complete a system to which this criteria is applied.

We refer to the information by the Bureau of Aeronautics, the Civil Aeronautics Board, and the Federal Aviation Administration that the flight control system is not reliable.

We believe that the use of a "fly-by-wire" flight control system is a step in the direction of a more reliable system for the control of aircraft in flight.

We hope this information will be helpful to you.

Sincerely,
For Memo
Dr. P. R. Kurzahls
Chief, Guidance Systems, GEP-100

DEPARTMENT OF TRANSPORTATION
FEDERAL AVIATION ADMINISTRATION

WASHINGTON, D.C. 20591

MAY 2 1974

Dr. Peter R. Kurzahls, Director
Guidance, Control & Information Systems, Code RE
NASA Headquarters
Washington, D.C. 20546

In the 1970s, companies began to develop “fly-by-wire” systems in which computers would control aircraft in flight. Developers asked the FAA what standards they would have to meet for certification. At first, the FAA just suggested that the flight control software should never be the cause of an accident – it should be as reliable as the wings of the aircraft. Wings are not supposed to fall off, and the flight control software must not fail. Failure causing an accident should be “extremely improbable” The developers felt they could not build or certify to such a standard and asked for some quantifiable goals they could attempt to meet.

FAA Reliability Requirements for Aircraft Fly-by-Wire Flight Control System - 1974

Section 25.1309 of the Federal Aviation Regulations requires that airplane systems be designed so that the occurrence of any failure condition (combinations of failures in addition to single failure considerations) which would prevent the continued safe flight and landing of the airplane is extremely improbable. The Federal Aviation Administration has accepted substantiating data for compliance with that requirement which shows by analysis that the predicted probability of occurrence of each such failure condition is 10^{-9} per hour of flight.

We further believe that failure of all channels on the same flight in a "fly-by-wire" flight control system should be extremely improbable; that is, be shown to have a probability of occurrence equivalent to that which has been shown for similar failure of all fully-powered hydraulic flight control systems on the same flight of an airplane with no manual back-up.

The FAA eventually came back with a requirement that the failure rate should by analysis be no more than 10^{-9} per hour of flight. The longest flight being estimated at 10 hours, this meant a failure only one time in 100,000,000 flights. And developers accepted that standard and built to it.

The certification of flight control software is today governed by a regulation called "DO-178C", (following 178A and 178B), published in 2012. The next slides give a quick overview of what's required for this certification.

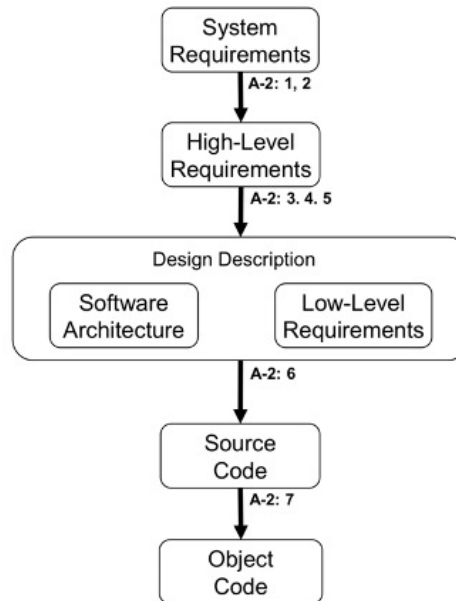
Certification of Flight Control software

- DO-178C
 - Published 2012, replacing prior DO-178B
 - "acceptable means, but not the only means, for showing compliance with the applicable airworthiness regulations for the software aspects of airborne systems and equipment certification."
 - Enables the use of "formal methods" for software assurance
 - Who inspects/certifies?
 - FAA Designees (on site rep of FAA, employed by aircraft company)



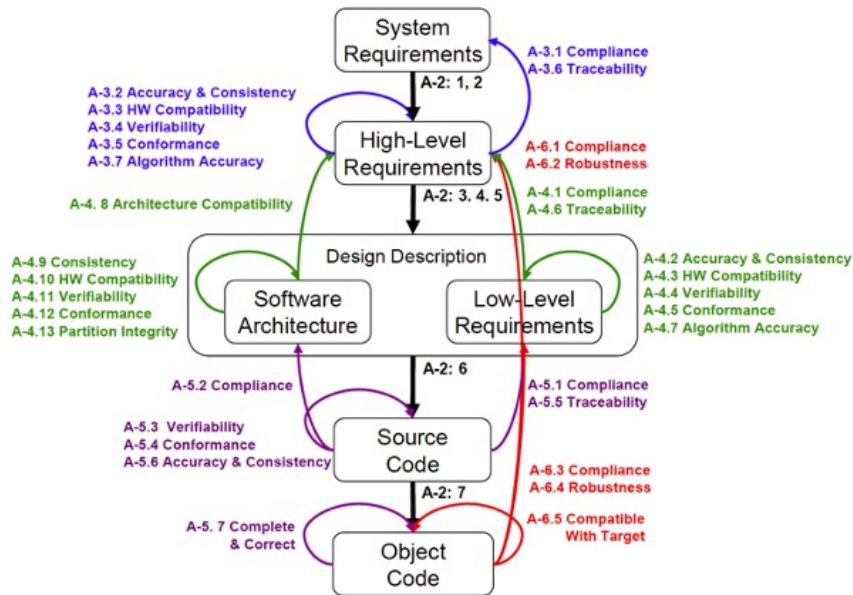
Critical flight control software is subject to stringent controls to assure that it won't contribute to a failure in flight. Since 2012, formal methods have been specifically called out as an acceptable means for showing compliance with airworthiness regulations for software aspects of airborne systems and equipment certification. The inspection and certification is largely done within industry facilities by people closely involved with the software development and testing as FAA "designees".

DO-178: Artifacts and Processes



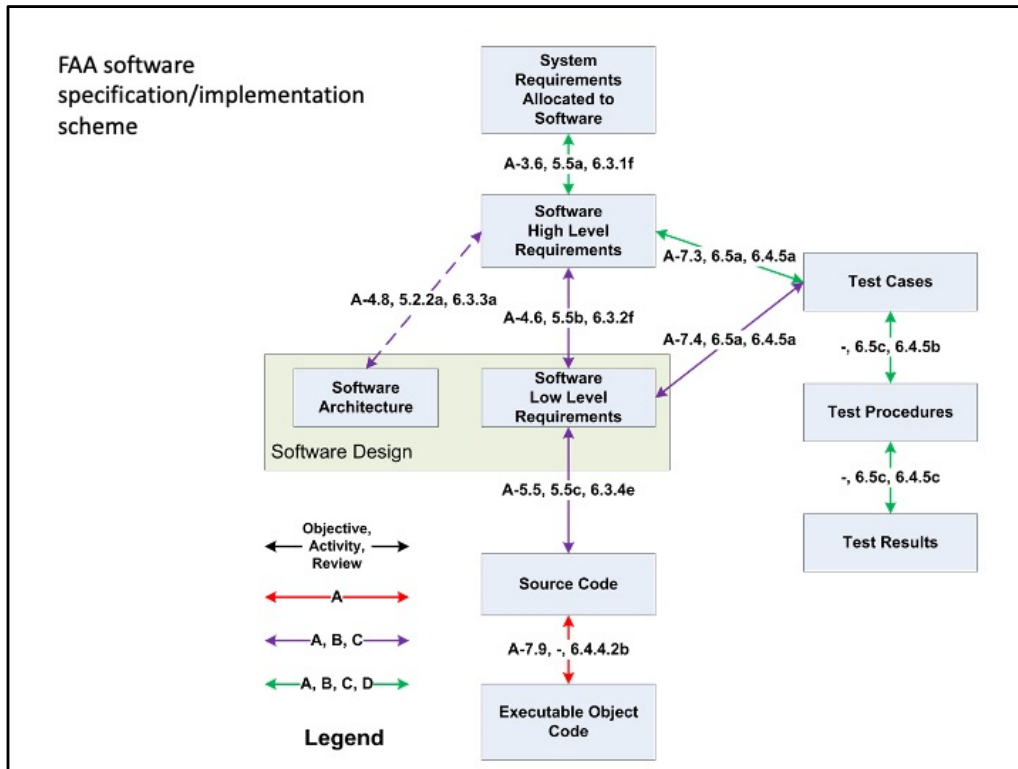
The software has to be documented in detail as shown here, from system requirements, through high level requirements and on down, all the way to the object code.

DO-178: Artifacts and Processes



The information contained within these slides was produced as part of NASA Contracts NNL14AA06C and NNL12AB85T.

This overlay shows some of the required mappings among the various levels of requirements, design, and implementation.



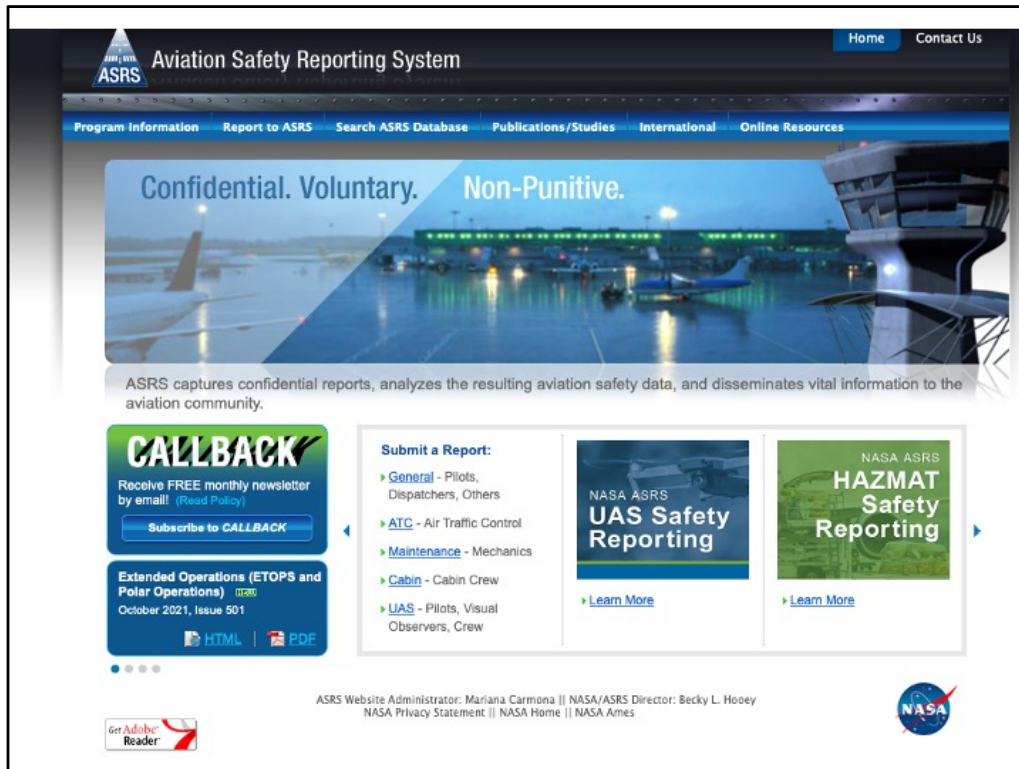
And here we see the documentation structure together with structure for testing.

FAA Designees

- FAA can delegate some of its inspection/certification duties to “designees”
- These may be employees of, e.g., Boeing
- Saves the FAA from having more employees, can save money
- But sets up potential conflict of interest

It was recognized years ago that the FAA might have a hard time recruiting and paying sufficient in-house staff to monitor aircraft manufacturers’ design and development processes. A scheme was devised whereby the manufacturer would provide designated personnel (“Designees”) who would act as representatives of the FAA and be on the premises of the manufacturer, assuring that FAA documentation and standards were being met.

This of course sets up a conflict of interest in that personnel being paid by the company are in effect charged with overseeing the people who are paying them. Nonetheless, the scheme has been in effect for decades and seemed to work.



After systems are designed, developed, and fielded there can of course still be accidents and also “near-misses” – events that might have turned out badly but didn’t by dint of good luck or good management. Accident and near miss reports can provide valuable feedback to both the FAA and industry on safety issues. But people can be reluctant to report such incidents because of concerns about personal and corporate liability and reputation.

This is the web page for the Aviation Safety Reporting System, which is widely considered a regulatory success – though its success comes in large part from being separated from regulation.

This is a system for anonymous reporting of “near misses” and other safety critical events. Identity of those reporting events is withheld, and by reporting, immunity is provided to the reporters.

It’s an FAA system, but it was set up and is operated by NASA, which has no regulatory role, and important difference.

The anonymity of the reporting and the separation of the reporting from the regulatory agency seem to be strong factors in the success of the system

The system was put in place in 1976; such a system had been suggested many years earlier

It provides a very valuable free online database that can be used by anyone for research.



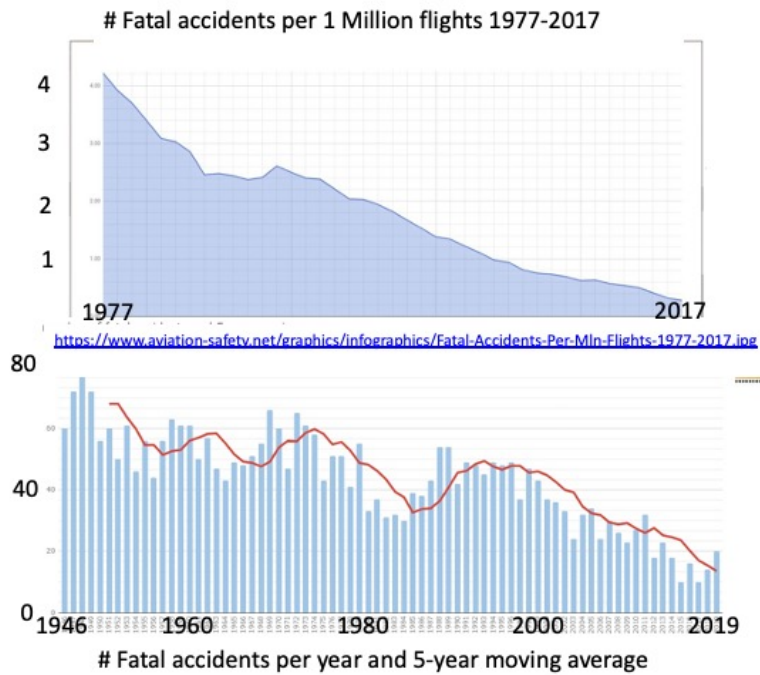
We can't discuss aircraft software regulation without addressing the recent terrible crashes of the Boeing 737 Max.

Boeing developed the MCAS (Maneuvering Characteristics Augmentation System), designed to compensate for larger, more forward engines without requiring pilot simulator training. The goal of MCAS is to detect an impending stall on takeoff and to correct it automatically by using the horizontal stabilizers to push the nose down. Unfortunately, it had a single point of failure. The angle of attack indicator is a sensor on the outside of the fuselage, subject to buffeting of the weather. The design called for redundant AoA sensors, one on each side of the fuselage as shown here, but the MCAS system did not compare the outputs of the two sensors to see if they were giving consistent readings. So failure of a single AoA sensor could trigger the MCAS system during takeoff.

Pilots could deal with this failure if they recognized what it was, but many had not been informed about the MCAS system and so didn't know how to respond in a chaotic situation.

Was this a software failure? Not evidently: software performed as specified and implemented. It was a design flaw that should have shown up in the analysis. Was this a failure of the "designee" system? Possibly so. The longer term effects on regulation are still to be determined.

How well has it worked?



Has safety regulation for airline aviation been effective? These charts taken from an airline safety website would seem to indicate it has been quite effective over a long period of time.

Aviation Regulation Summary



- **Sparks:** Accidents



- **Carrots:**

- Government certification of pilots, aircraft, and operation of air traffic control encourages **passenger trust** and provides some legal cover when failures occur
- ASRS (Aviation Safety Reporting System): anonymous/no-fault reporting has been effective: reports go to NASA rather than FAA, to preserve anonymity of reporters



- **Sticks:**

- controls on critical software development
- ability to ground aircraft,
- deny licenses,
- control flight operations



- **Cybersecurity?**

- Just getting started.

What are the sparks, carrots, sticks for aviation regulation?

Has this realm of regulation been successful? In terms of passenger safety and industry development, the answer has to be yes.

Has it addressed cybersecurity? For a long time, and like many other domains, security was really handled by the fact that only trusted individuals had access to the systems and the opportunity for hacking them was minimal.

But that situation has changed somewhat in recent years, and the FAA has begun to respond.

By fall 2019, the FAA began to pay attention to explicit cybersecurity testing of actual aircraft

And the FAA changed its organization; the Air Traffic Operations (ATO) organization at FAA now has a Cybersecurity Group (ACG)

This group appears focused on assuring cybersecurity of FAA's operational systems (vs. certifying aircraft cybersecurity)

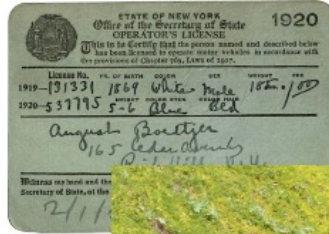
Oct. 2020, GAO called for strengthening FAA oversight of avionics cybersecurity. It's not clear to me whether GAO's recommendations have been implemented as yet, but it appears that DHS and DoT (FAA) are collaborating.

Key Takeaways from U.S. Aviation Regulation

- Regulation grew up with the industry and largely with industry cooperation because of common interest in safety
- Mechanisms supporting anonymous reporting of problems (“near misses”) (ASRS) and for investigating and understanding the reasons for accidents have generally worked well
- As complexity of technology grew, centralized review of technical details (e.g. flight critical software) was seen as impractical
 - FAA developed mechanisms to place trust in company personnel for certain review functions
- This worked well as long as companies maintained a strong “safety culture” internally
- Recent events are calling this approach into question
- Cybersecurity is a relatively new concern; not clear how this will play out yet

Again, summarizing.

History of Automobile Regulation



1949 Nash – seatbelt available

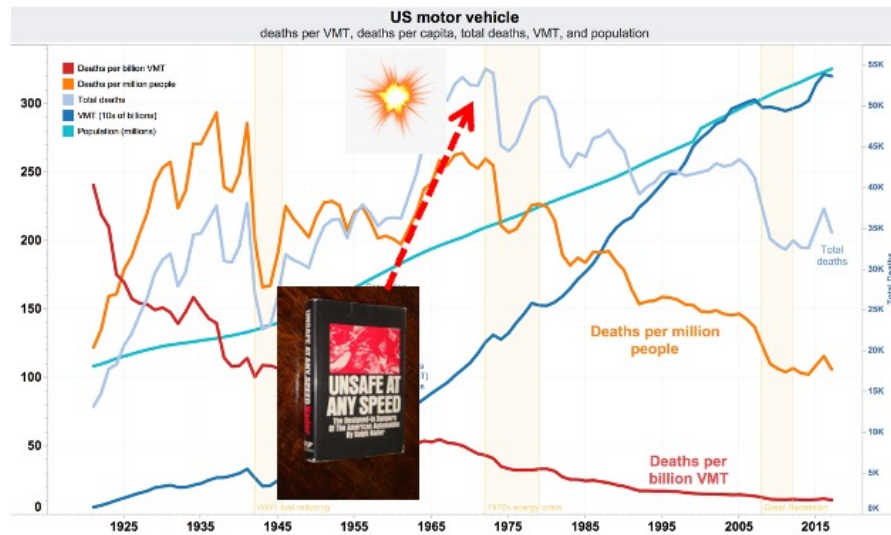
1959 Nils Ohlin invents 3-point belt for Volvo



There was little or no federal regulation of automobiles in the US for a long time. State licensing of drivers started in the early 1900s (1920 New York drivers license on the slide); South Dakota, the last state to require a license, held off until 1954. Early state laws regulated speed and drunk driving. Traffic fatalities became a matter of public concern beginning in the mid-1930s, and at that time several safety minded individuals began pressing the industry to include better safety features in its designs – to pad dashboards, remove protruding knobs, install seat belts, but to little effect. The industry as a whole resisted, favoring the position that accidents were caused by drivers, and so the focus should be on driver training and licensing, not on safe car design. Their view seemed to be that incorporating safety features might discourage customers by suggesting that driving could be dangerous. By the mid 1950s, some parts of the industry began to advertise safety as a feature of their designs. Lap belts, (first US patent 1885), were first offered on Nash cars in 1949 (that's a 1949 Nash on the slide), as an option, but were rarely purchased. Air bags were invented in 1951. Nils Bohlin, a Volvo engineer experienced in designing ejection seats for aircraft, designed the 3-point

lap belt in 1959, and Volvo not only began shipping cars with them, it made the patent available for free to other companies.

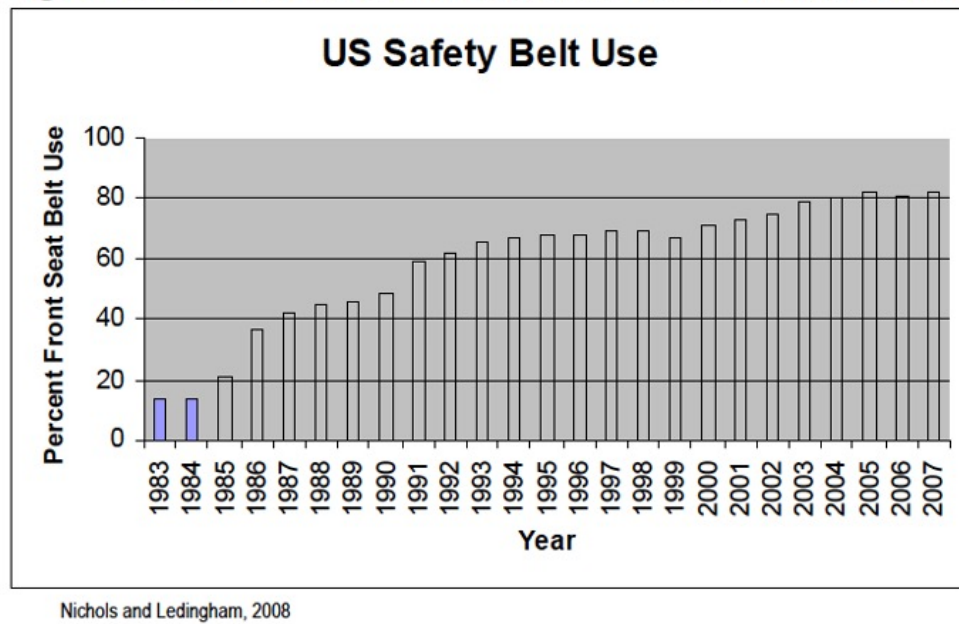
Motor Vehicle Fatalities by year



Source: https://en.wikipedia.org/wiki/Motor_vehicle_fatality_rate_in_U.S._by_year

This chart shows the US total fatalities and fatalities per mile driven from 1920 to 2018. Note that the publication of Ralph Nader's book "Unsafe at any speed" and his subsequent Congressional testimony coincided with a peak in total fatalities in the late 1960s. The public outcry led to the Motor Vehicle Safety Act (1966), which directed the Commerce Secretary to create motor vehicle safety standards. This responsibility moved to the Department of Transportation when it was created in 1968, and subsequently the Highway Safety Act created the National Highway Traffic and Safety Administration (NHTSA) in 1970. NHTSA remains the body in charge of Federal Motor Vehicle Safety Standards (FMVSS). Seat belts, or seat belt anchors, were mandated in new cars in (1968). Nevertheless, they were not widely used until states began passing laws to mandate their use ("click it or ticket"). How this came about is an interesting story.

Figure 1. Observed Seat Belt Use Rates in the United States, 1983-2007



This chart of seat belt use in the U.S. only starts in 1983, and in that year it was well under 20%. Lap and shoulder belts were mandated in 1968. In 1974, NHTSA mandated a seat belt interlock, so that a car would not start until the driver buckled up. The public revolted at this and at the annoying buzzer that went off until the belt was fastened. Congress responded quickly, killing the interlock requirement and limiting the buzzing to 8 seconds. NHTSA didn't give up. In 1977, it proposed a rule that automakers include some kind of passive restraint mechanism requiring no action on the part of occupants that would protect them in a 35 mph collision with a brick wall. The only options at the time were the largely untried air bag systems and systems of "automatic" seat belts that fastened automatically when the door was closed. Before the rule could take effect, Ronald Reagan was elected on a platform of deregulation, including of the auto industry, and promptly rescinded the NHTSA rule requiring passive restraints. Insurance companies sued, and the Supreme Court ruled unanimously that the revocation of the rule was "arbitrary and capricious" and rescinding it had been improperly done. Elizabeth Dole, Secretary of Transportation, had to find a way out. In 1985, Dole issued a rule that automakers had to install passive restraint systems (airbags or automatic seat belts) unless states representing two thirds of the US population enacted laws to make seat belt use mandatory before April 1, 1989. The auto industry generally favored the seat belt systems, since they were already largely in place and were much cheaper than air bag systems. So the auto industry immediately lobbied all the state legislatures to pass mandatory seat belt laws, and many states did --- but not enough. So in the end, passive restraint

systems were required, and in 1998, air bags were required. But the side effect of mandatory “click it or ticket” laws dramatically improved the rate of use of seat belts across the country, as the graph shows. A 2008 NHTSA study showed that the difference in usage rates between states with higher and lower usage rates was largely due to drivers’ perceived likelihood of getting a ticket – so mandates and enforcement can be effective. Today, only New Hampshire does not have a law mandating seat belt use, and in 2020 the nationwide usage rate was over 90%. And both overall fatalities and fatality rates per mile driven, as shown on the earlier chart, have declined substantially from where they were in the mid-1960s.

Automotive Safety Summary



- Sparks: public concern over traffic fatalities, Ralph Nader's book and testimony



- Sticks: manufacturing requirements, enforced mandates, lawsuits



- Carrot: attract safety-minded customers, avoid safety recalls, avoid mandates

I think it's fair to say that automotive regulation has relied much more on the stick than the carrot



In the 1950s and 1960s awareness of pollution and its effects on the environment grew. Smog had already been an issue in California for a long time. Air pollution and cars were first linked in the early 1950's by a California researcher who determined that pollutants from traffic were to blame for the smoggy skies over Los Angeles. At the time, typical new cars were emitting nearly 13 grams per mile hydrocarbons (HC), 3.6 grams per mile nitrogen oxides (NO_x), and 87 grams per mile carbon monoxide (CO). Rachel Carson's publication of *Silent Spring* in 1962 put a spotlight on the effects of pesticides in the environment, including the effects of DDT on birds. In June 1969, a river in Ohio actually caught fire, damaging a nearby bridge. All of this and more prompted President Nixon to sign the National Environmental Policy Act on January, 1, 1970, creating a Council on Environmental Quality to, among other things, advise him on how to organize the government to address national environmental needs. He also sent Congress a message requesting action on a broad range of environmental issues, including automobile pollution. This led to the creation of the Environmental Protection Agency in December, 1970, and subsequent passage of the 1970 US Clean Air Act later that month.

Another major player in auto emissions regulation is the California Air Resources Board (CARB), which was created three years earlier when then Governor Ronald Reagan signed legislation combining two existing pollution control organizations in the California government. Over the years, California's emissions regulations, often stronger than the Federal ones, have had significant sway with automobile manufacturers. Most recently, President Trump aimed to roll back EPA's mileage

regulations, but the CARB was able to negotiate with several major auto builders to meet more their more stringent standards anyway.

Over the years, EPA banned the use of lead additives to gasoline and made rules requiring increases in automobile miles per gallon and decreasing levels of tailpipe emissions.

<https://www.epa.gov/transportation-air-pollution-and-climate-change/timeline-major-accomplishments-transportation-air>



VW Diesel Emission scandal



Fast forward to 2006. VW had a problem. High-mileage diesel engines had difficulty meeting standards for reducing emissions of Nitrous Oxides (NOx). They thought they had a cheap solution, but it didn't work in practice. So, instead of investing in more expensive technology to clean the exhaust, they decided to game the system. They introduced engine control software that would run the engine in a low pollution mode when it detected the conditions under which the EPA tested the engine, but in a different, high pollution, high mileage mode, when actually driven on the road by customers. This "defeat device" was in effect a Trojan Horse in the engine control software.

This scheme was effective for several years (witness the 2009 photo on the slide). But it unravelled when a West Virginia University researcher, under contract to the California Air Resources Board, measured actual tailpipe emissions on the road in 2014. The subsequent scandal has cost VW billions of dollars in fines and lost reputation, but they remain one of the top two auto manufacturers in the world.

Automotive emission summary

- Software-based products introduce hard-to-detect possibilities for fraud
- Observing product behavior in the real world is the acid test
- But it may be necessary to gain visibility into the software during development for high assurance

Automotive regulation takeaways

- Auto industry grew up without regulation and has in general resisted regulation in all domains
- Nevertheless, regulators have achieved significant gains in both safety and emissions control
- The threat of regulation can be as motivating for industry as an actual regulation

Computer/cybersecurity regulation: The situation in the late 1970s/early 1980s



The military used computers from day one. In the beginning, it often fabricated its own machines, but by the late 1960s, it was using large scale timesharing systems built commercially. It wanted to believe that the controls built into the operating systems for these machines could enforce military security policies, but numerous penetration exercises demonstrated otherwise. Early studies suggested that a new OS architecture built around the concept of a “reference monitor” might achieve the elusive goal of multilevel security. Prototypes were constructed, but getting security into the commercial market place was a challenge. As computers shrank from the room-size Univac 1108 to the PDP-11 minicomputer and then the more capable mid-size VAXes of the late 1970s, operating systems evolved as well; Unix was well established by 1980. At that time, some in the Defense Department (specifically, Steve Walker) recognized DoD would not dominate the computer market and so could not dictate to producers what to build. But they also believed that products with better security would not cost more to manufacture, once developed, and the security features they wanted could live within the commercial products and not interfere. So the trick

was to persuade the companies to invest in better security even though the primary market didn't seem to care.

The Strategy

The basic idea:



- provide “Consumer Reports” like information about the security (or lack thereof) of computer systems



- provide a market for more secure computers by requiring DoD systems to procure computers that did well in the ratings

Intended result: gradual improvement in built-in security in commercial products, availability of some high assurance systems for government needs

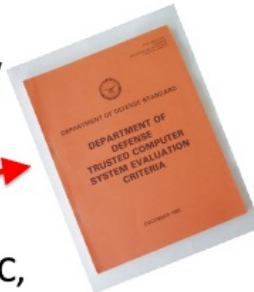
Precedent: procurement of crypto gear by NSA

- Endorsed Crypto Products List
- Preferred Products List (PPL) [TEMPEST]
- Degausser Qualified Products List

Implementing the strategy

- Needed:

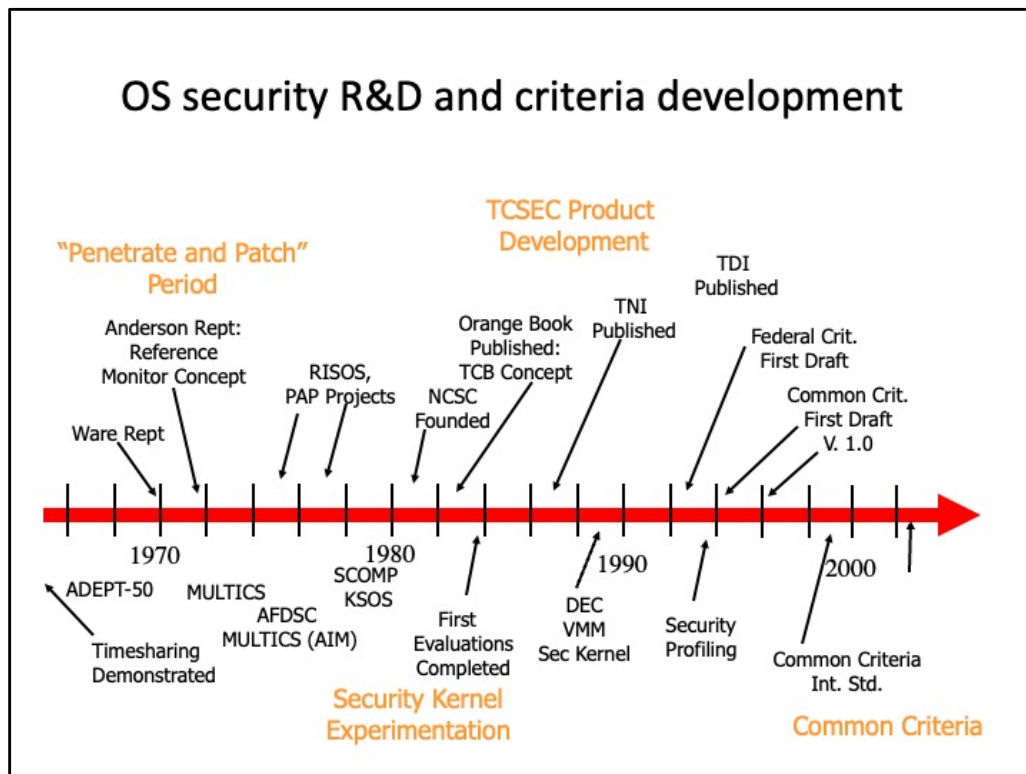
- basis for evaluations
- organization to do the evaluations: NCSEC, subsequently NCSC, originally aimed at NBS (NIST), landed at NSA
- industry cooperation to submit systems for evaluation



- More carrots for industry:



- Only systems submitted voluntarily would be evaluated
- Government would bear the evaluation cost (though vendor must produce evidence required)



This timeline covers development of computers and operating systems with security objectives as well as the development of the criteria and organizations for evaluating them.

Note these particular dates:

1981: NCSEC/NCSC established

1983: initial evaluation criteria published

1984: initial evaluations completed: RACF-MVS--C1 ACF2-MVS/SP--C2

SCOMP- A1

2000: Last TCSEC evaluation completed: Sybase Adaptive Server Anywhere, v.7.0.0 Sept. 2000 [C2]

Total of 85 evaluation certificates issued; 29 more initiated but not completed (withdrawn?)

As of 2002: in US, 6 firms are in business doing evaluations under the Common Criteria

Other countries were interested in evaluating security as well, and this led to the globalization of these efforts,

UK developed its own criteria but farmed evaluations out to Commercial Licensed Evaluation Facilities (CLEFs), and there were German, French and finally harmonized European criteria

In an effort to provide a more flexible structure better suited to the way computing architectures had developed, the “Common Criteria” which split function and assurance was developed.

There are now many laboratories worldwide that will perform these evaluations; there’s a list at <https://www.commoncriteriaportal.org/labs/>

16 countries have them; seven are in the US (four in the DC area, two in Austin, one in California)

Why it didn't work as planned

- Evaluations took too long
 - “criteria creep”
 - vendor drag - slow response
 - re-evaluation needed of new releases, hardware [RAMP]
- Inadequate procurement pull
 - not enough evaluated products to assure competitive procurements
 - perverse incentives for uncompleted evaluations
 - systems, not products, were procured
 - no mechanism to evaluate “GOTS” products
 - networking omitted
- Structure of the criteria vs. products in the market
 - bundling of features and assurance requirements

Talk through the points on the slide

Some lessons

- Information about product security properties is hard to obtain
 - difficult to quantify and assess
 - time consuming to obtain with confidence
 - unstable in the face of system changes
- Customers have historically preferred newer systems with more features and less certain security properties to older systems with fewer features and better-established security properties

Talk through the points on the slide

Cryptography Regulation

- Late 80's – early 90's:
 - Export control – 40 bit keys, later 64-bit
 - Initial security for 802.11 “WEP” had 64 bit keys
 - So no one took the protocol analysis too seriously
 - Subsequently relaxation of controls enabled longer keys – but latent flaws in the initial WEP protocol were then uncovered

How might these models help in reducing vulnerabilities in software?

- Building codes for software?
- NTSB equivalent for cyber incidents?
- Liability for final product assemblers?

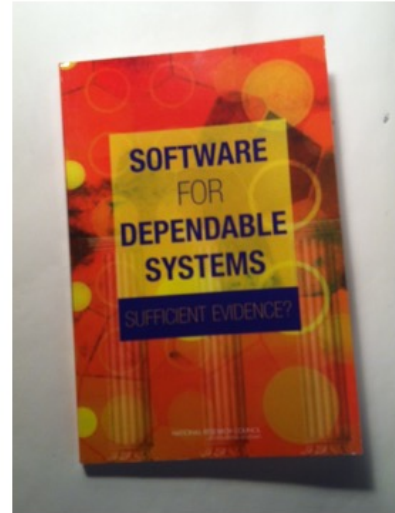
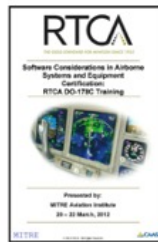
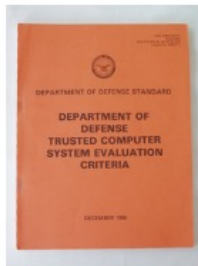
So can we learn anything useful from these examples?

How can we incentivize the production of systems with fewer vulnerabilities, systems that can realistically be defended without a constant series of patches and corrections?

Here are a few ideas. I hope you will pursue these or generate more that are relevant to the domain in which you work.

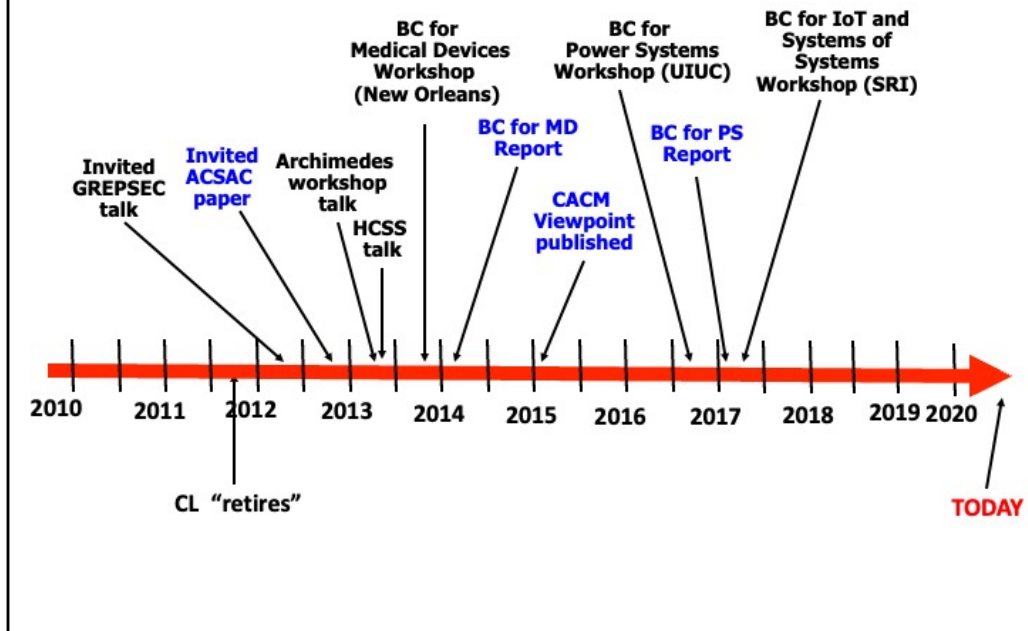
What about a building code for critical software systems?

- What can we require?
- What can we inspect/test?
- What do previous examples teach us?



In many respects this isn't a new idea: TCSEC and other examples are out there. But some of them haven't worked very well. We need not to make the same mistakes.

BCs for BC Chronology



IEEE CYBER SECURITY

All reports available at:
<https://cybersecurity.ieee.org/center-for-secure-design/>

Building Code for Medical Device Software Security

Tom Haigh and Carl Landwehr

IEEE IEEE@computer society

IEEE CYBER SECURITY

Building Code for Power System Software Security

Carl E. Landwehr, Cyber Security and Privacy Research Institute, CPSR, George Mason University
Alfonso Valdes, Information Trust Institute (ITI), University of Illinois

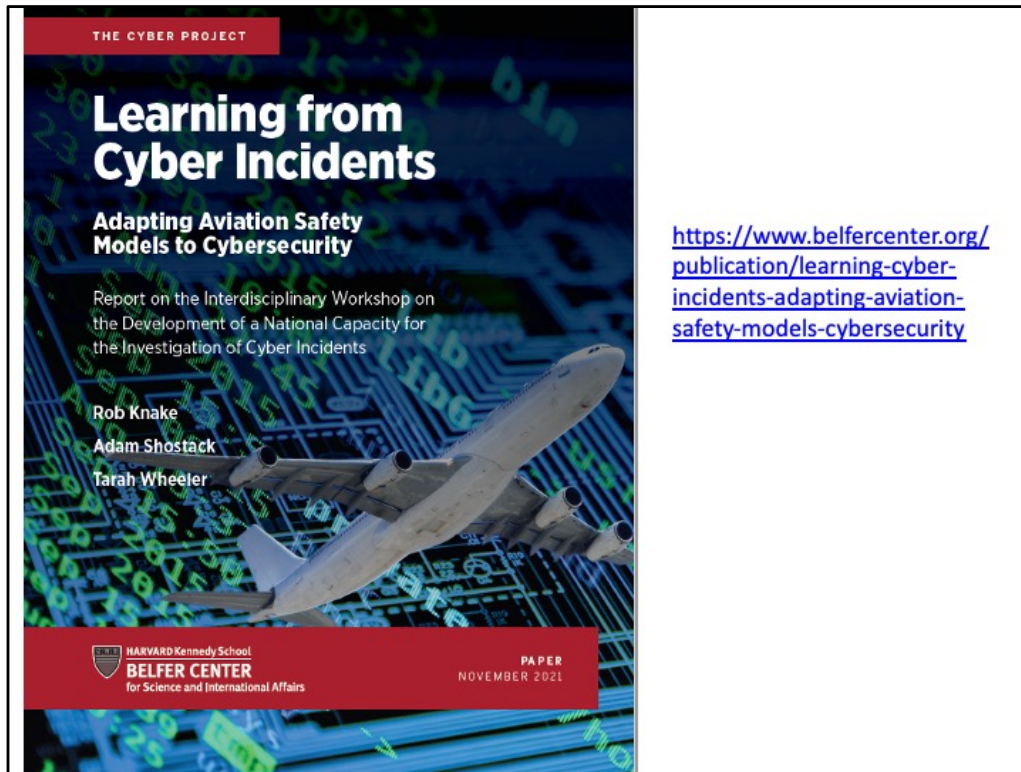
IEEE IEEE@computer society

IEEE CYBER SECURITY

Building Code for the Internet of Things

Ulf Lindqvist and Michael Locasto

IEEE IEEE@computer society READ ON



Since 1991, people have suggested a Cyber Incident Investigation Board to investigate cybersecurity incidents

Last week, Harvard's Belfer Center published the first serious attempt to see how such a board might operate:

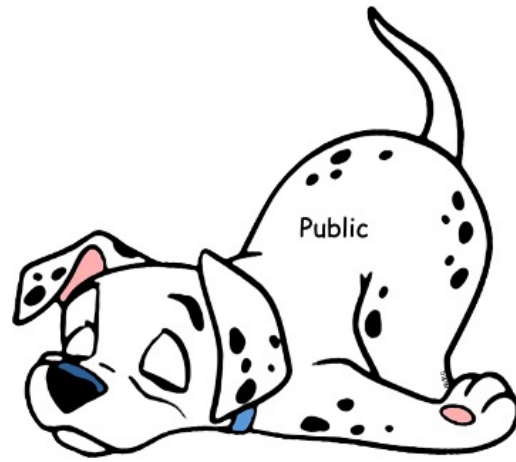
Some positive signs

- Cyberspace Solarium Commission recommendations gaining traction
- Biden May 2021 Executive Order 14028 on improving cybersecurity
- Consumer's Reports continuing to pursue security/privacy ratings
- UL providing tools for security evaluation

Summing up ...

1. The public is like a dog that likes to sleep.
2. When nagged, the dog can be aroused.
3. The tail won't wag the dog
4. The trick is to set up the incentives so that regulators aren't overwhelmed with enforcement actions and industry develops solutions that benefit society as a whole.

The public is like a dog



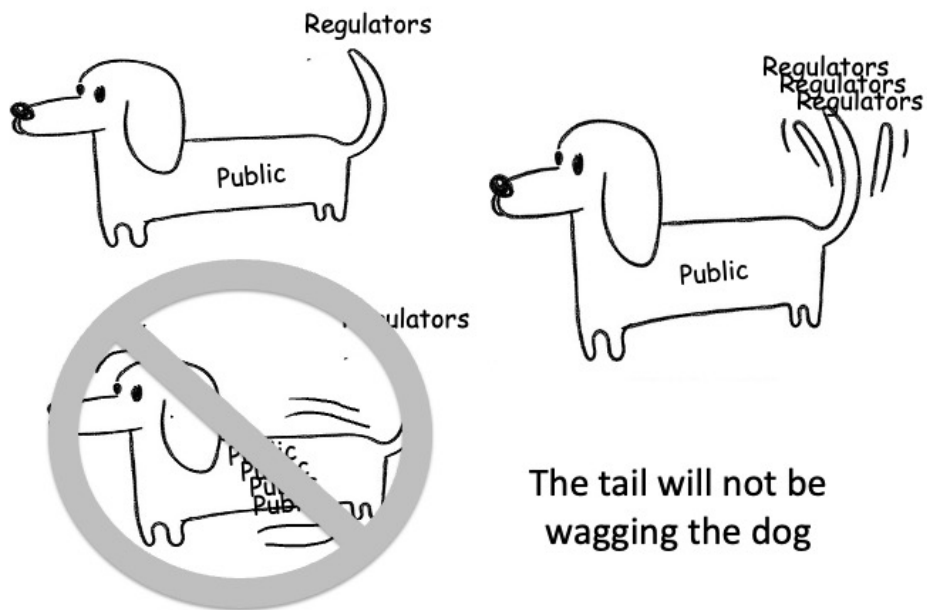
... that prefers to sleep



But it can be aroused

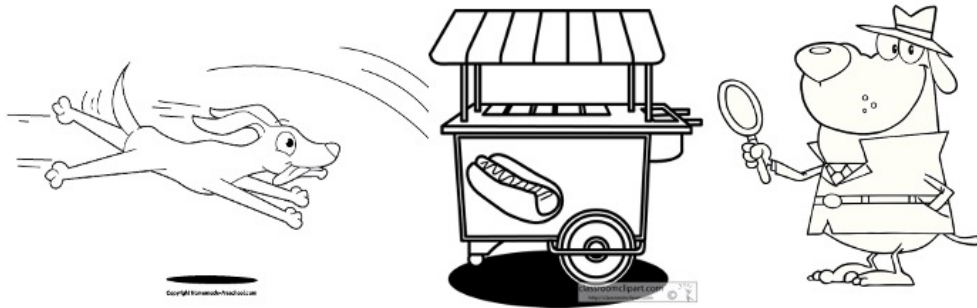
Fleas can wake the dog up if they bite long enough and widely enough.

And if it is...



The regulator is only on the tail on the dog. When the dog wakes up it may wag its tail.
The tail will not be wagging the dog

**With the right incentives, industry will develop
products meeting broad public needs**



But continuing vigilance is required

Dogs will flock to the hot dog vendor, but we better be sure the hot dogs are safe as well as tasty.

The talk on one slide



- Technology has been regulated for a long time
- Regulations typically involve a spark, a carrot, and a stick
- How these elements are crafted can make a big difference in the effectiveness of a regulation
- The evolution of the software and digital systems marketplace has not favored investment in high-assurance software development, except where the public has demanded it
- Developing the incentives for high assurance development for critical devices and systems should be a focus for both regulators and software professionals