Future challenges in security engineering

Ross Anderson

Cambridge and Edinburgh

How does IoT change safety?

- The EU regulates safety of all sorts of devices
- In 2015, they asked Éireann Leverett, Richard Clayton and me to examine what IoT implied
- 2016 report (WEIS 2017): once there’s software everywhere, safety and security get entangled
- (The two are the same in most EU languages– sicurezza, seguridad, sûreté, Sicherheit, …)
- How will we update safety regulation (and safety regulators) to cope?

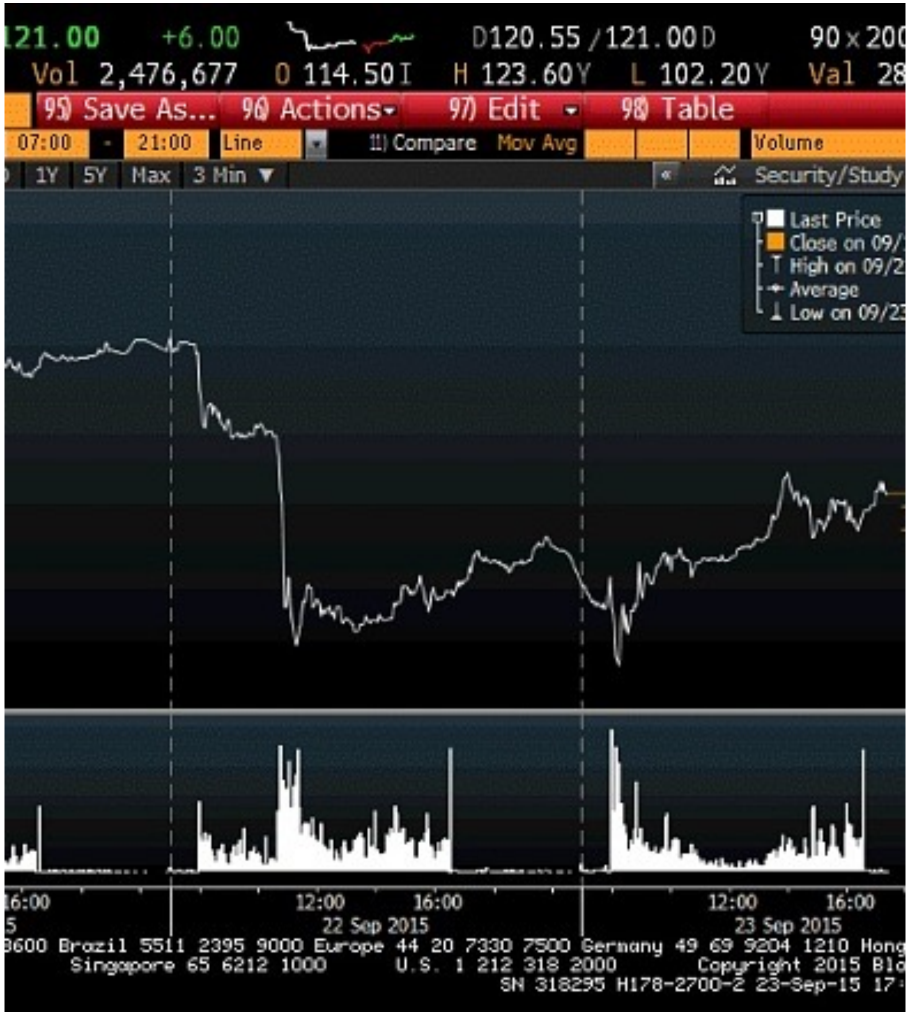
Safety engineering

- Markets do safety in some industries (aviation) way better than others (medicine)
- Cars were dreadful until Nader’s ‘Unsafe at Any Speed’ led to the NHTSA
- In the EU, we have broad frameworks such as the Product Liability Directive (all goods), sectoral laws such as a Directive on type approval for cars, plus many detailed rules
- Over 20 EU agencies (plus UNECE) in play

When cars get hacked (2)

- 2011: Carshark needed physical access
- 2015: Charlie Miller and Chris Valasek hacked a jeep Cherokee via Chrysler’s Uconnect
- So now we just need your IP address!
- Suddenly people cared…
- Chrysler recalled 1.4m vehicles for software fix

When cars get hacked (3)


121.00 +6.00
D120.55 / 121.00 D
90 x 200
Vol 2,476,677 O 114.50 I H 123.60 Y L 102.20 Y Val 28
95) Save As... 96) Actions 97) Edit 98) Table
07:00 - 21:00 Line
11) Compare Mov Avg
Volume
1Y 5Y Max 3 Min
Security/Study
Last Price
Close on 09/
High on 09/2
Average
Low on 09/23
16:00
12:00
16:00
22 Sep 2015
12:00
16:00
23 Sep 2015
3600 Brazil 5511 2395 9000 Europe 44 20 7330 7500 Germany 49 69 9204 1210 Hong
Singapore 65 6212 1000 U.S. 1 212 318 2000 Copyright 2015 Blo
SN 318295 H178-2700-2 23-Sep-15 17

P Bag Vol: 250ml
Volume Left 246.9
A Infused 3.1
Press OK to start

1 SELECT mg/mL
2 SELECT µg/mL
3 SELECT mL

Epidural 11:35 90%
A:Bupivicaine/Fentanyl
500ml
Rate :
0 mL/hr
Pt Bolus 0
mmHg 0 720

Medical Devices

- Research by Harold Thimbleby: hospital safety usability failures kill about 2000 p.a. in the UK, about the same as road accidents
- Safety usability ignored – incentives wrong…
- But attacks are harder to ignore – Kevin Fu’s Wi-Fi tampering demo in 2015 led the FDA to blacklist the Hospira Symbiq infusion pump
- 2017: recall of 450,000 St Jude pacemakers
- We were asked: what should Europe do?

Medical Devices (2)

- The Medical Device Directives have been revised: from 2021 it requires post-market surveillance, a per-device risk management plan, ergonomic design …
- Reg 17.2: ‘for devices that incorporate software… the software shall be developed … in accordance with the state of the art taking into account the principles of development life cycle, risk management, including information security, verification and validation’

Medical Devices (3)

- 18.8 ‘Devices shall be designed and manufactured in such a way as to protect, as far as possible, against unauthorised access that could hamper the device from functioning as intended’.
- It’s still not perfect (there’s wriggle room on ergonomics, network security assumptions…) but it’s a huge improvement!

Industrial Control Systems

- Electricity substations: 40-year lifecycle, protocols (DNP3) don't support authentication
- IP networking: suddenly anyone who knows a sensor's IP address can read from it, and with an actuator's IP address you can activate it
- Ten years ago, we found the only practical fix was to re-perimeterise!
- Have a firewall and replace it every 5 years
- But then there were smart meters: 'Who controls the off switch?'

Broad questions include…

- Who will investigate incidents, and to whom will they be reported?
- How do we embed responsible disclosure?
- How do we bring safety engineers and security engineers together?
- Will regulators all need security engineers?
- How do we prevent abusive lock-in? Tech is plagued by monopolies large and small…

Our recommendations included

- Requiring vendors to certify that products can be patched if need be
- Requiring a secure development lifecycle with vulnerability management
- Cybersecurity advice body for European safety regulators
- Duty to report breaches and vulnerabilities to safety regulators and users
- Extending product liability to services

The punch line

- Phones, laptops: patch them monthly, but make them obsolete quickly so you don’t have to support 100 different models

The punch line

- Phones, laptops: patch them monthly, but make them obsolete quickly so you don’t have to support 100 different models
- Cars, medical devices: we test them to death before release, but don’t connect them to the Internet, and almost never patch

The punch line

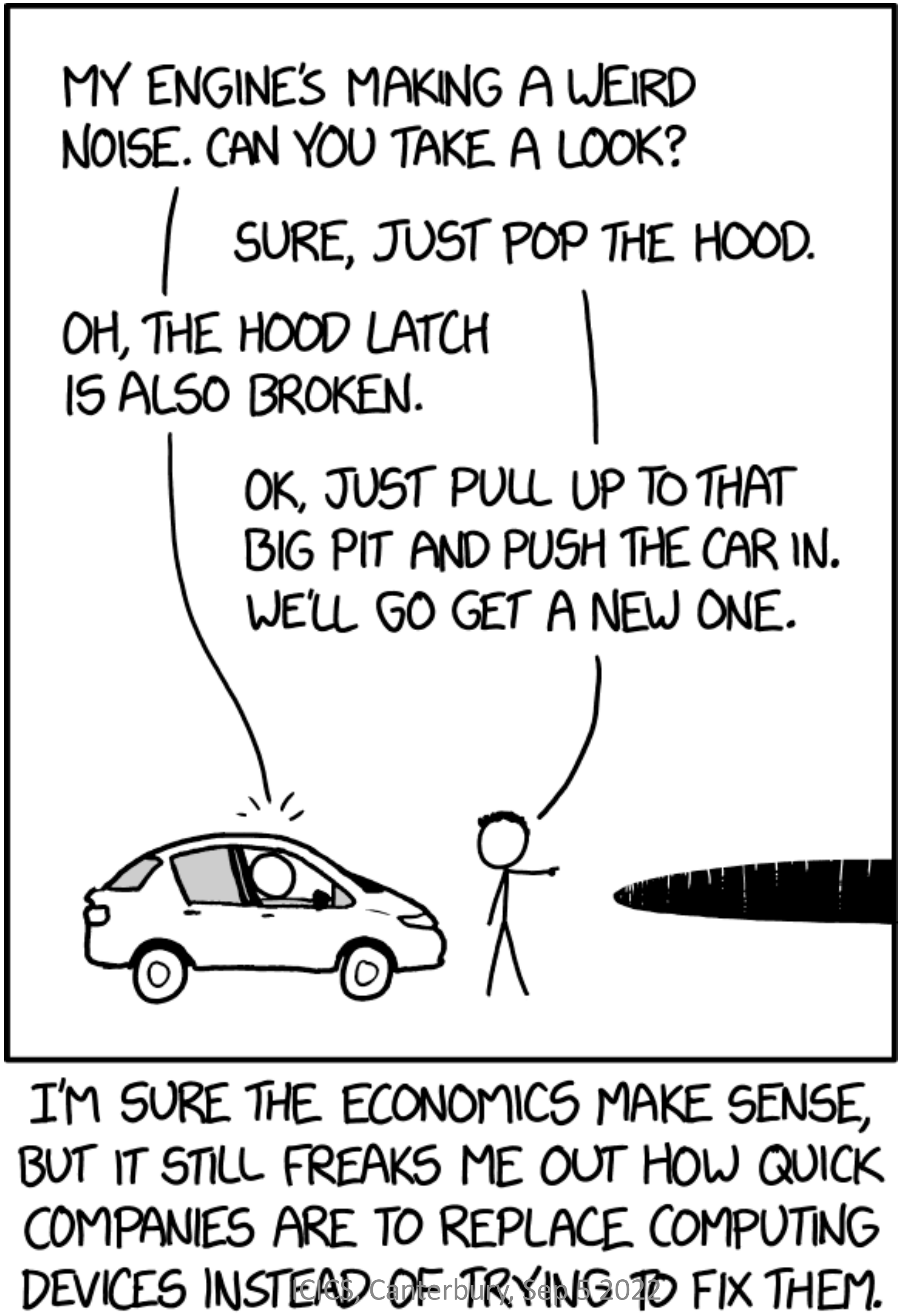
- Phones, laptops: patch them monthly, but make them obsolete quickly so you don’t have to support 100 different models
- Cars, medical devices: we test them to death before release, but don’t connect them to the Internet, and almost never patch
- So what happens to support costs now we’re starting to connect all sorts of durable goods to the Internet, and have to patch them?

The trilemma

- Standard safety lifecycle, no patching -> safety + sustainability -> go online, get hacked
- Standard security lifecycle, patching -> breaks safety certification
- Patching plus redoing safety certification with current methods -> costs of maintaining safety rating can be sky high
- So: can we get safety, security and sustainability at the same time?

Vehicle lifecycle economics

- Vehicle lifetimes in Europe have about doubled in 40 years
- Average age at scrappage in UK now 14.8y
- Some vehicle makers wanted to say “scrap it after 6 years and buy a new one!”
- But the embedded CO_2 cost of a car often exceeds its lifetime fuel burn
- And what about Africa, where most vehicles are imported second-hand?


MY ENGINE'S MAKING A WEIRD NOISE. CAN YOU TAKE A LOOK?
SURE, JUST POP THE HOOD.
OH, THE HOOD LATCH IS ALSO BROKEN.
OK, JUST PULL UP TO THAT BIG PIT AND PUSH THE CAR IN. WE'LL GO GET A NEW ONE.
I'M SURE THE ECONOMICS MAKE SENSE, BUT IT STILL FREAKS ME OUT HOW QUICK COMPANIES ARE TO REPLACE COMPUTING DEVICES INSTEAD OF TRYING TO FIX THEM.

The economics of dependability

- Complex socio-technical systems often fail because of poor incentives
- If Alice guards the system but Bob pays the cost of failure, you can expect trouble!
- Security economics explains platform security problems, the patching cycle, liability games and much else that we used to treat as just bad luck
- The same principles apply to safety and safety and security are becoming entangled

2019 Consumer Protection Upgrade

- 2019/771: EU directive on Sales of Goods
- Buyers of goods with digital elements are entitled to necessary updates for two years, or for longer if this is a reasonable expectation of the customer
- Trader has burden of proof in first two years
- But what is ‘a reasonable expectation of the customer’?

What’s a reasonable expectation?

- Cars: maybe 20 years (3 R&D, 7 retail, 10 years from last instance leaving the showroom)
- Domestic appliances: 10 years spares obligation, plus store life … 15?
- Medical devices: if a pacemaker has a 10-year in-service life, then surely 15 or 20?
- Electricity substations: maybe 40 years
- WEF “circular vision for electronics”

The grand challenge for research

- If the durable goods we’re designing today are still working in 2060, things must change
- Computer science = managing complexity
- The history goes through high-level languages, then types, then objects, and tools like git, Jenkins, Coverity …
- What else will be needed for sustainable computing once we have software in just about everything?

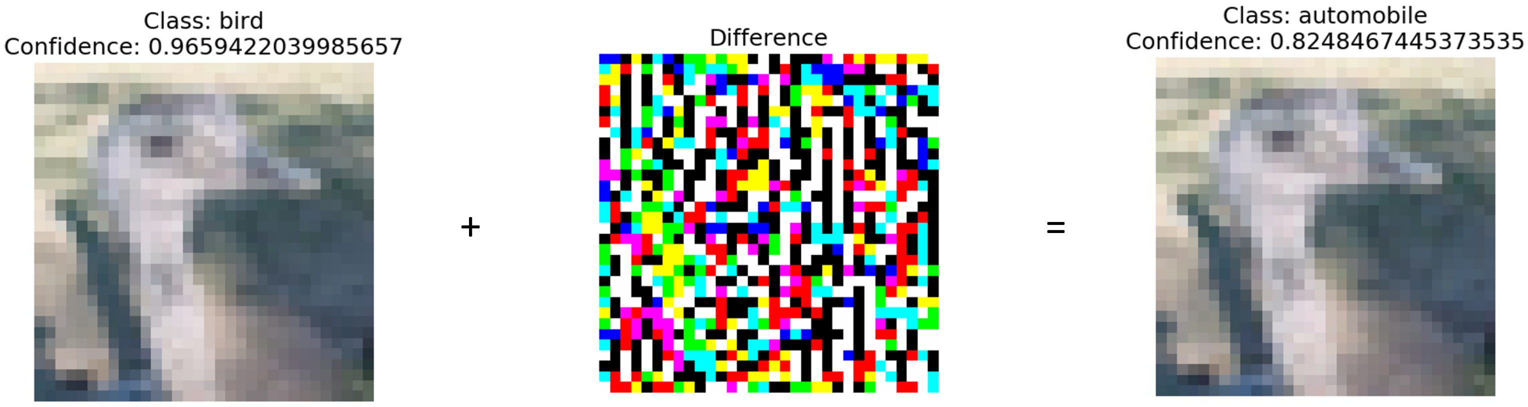
Effects on research and teaching

- Since 2016–7 I’ve been teaching safety and security engineering in the same course to first-year undergraduates (now all online!)
- We started to look at what we can do to make the tool chain more sustainable
- For example, can we stop compiler writers opening up timing channels?
- Better ways to communicate intent might help (see “What you get is what you C”)

Effects of machine learning

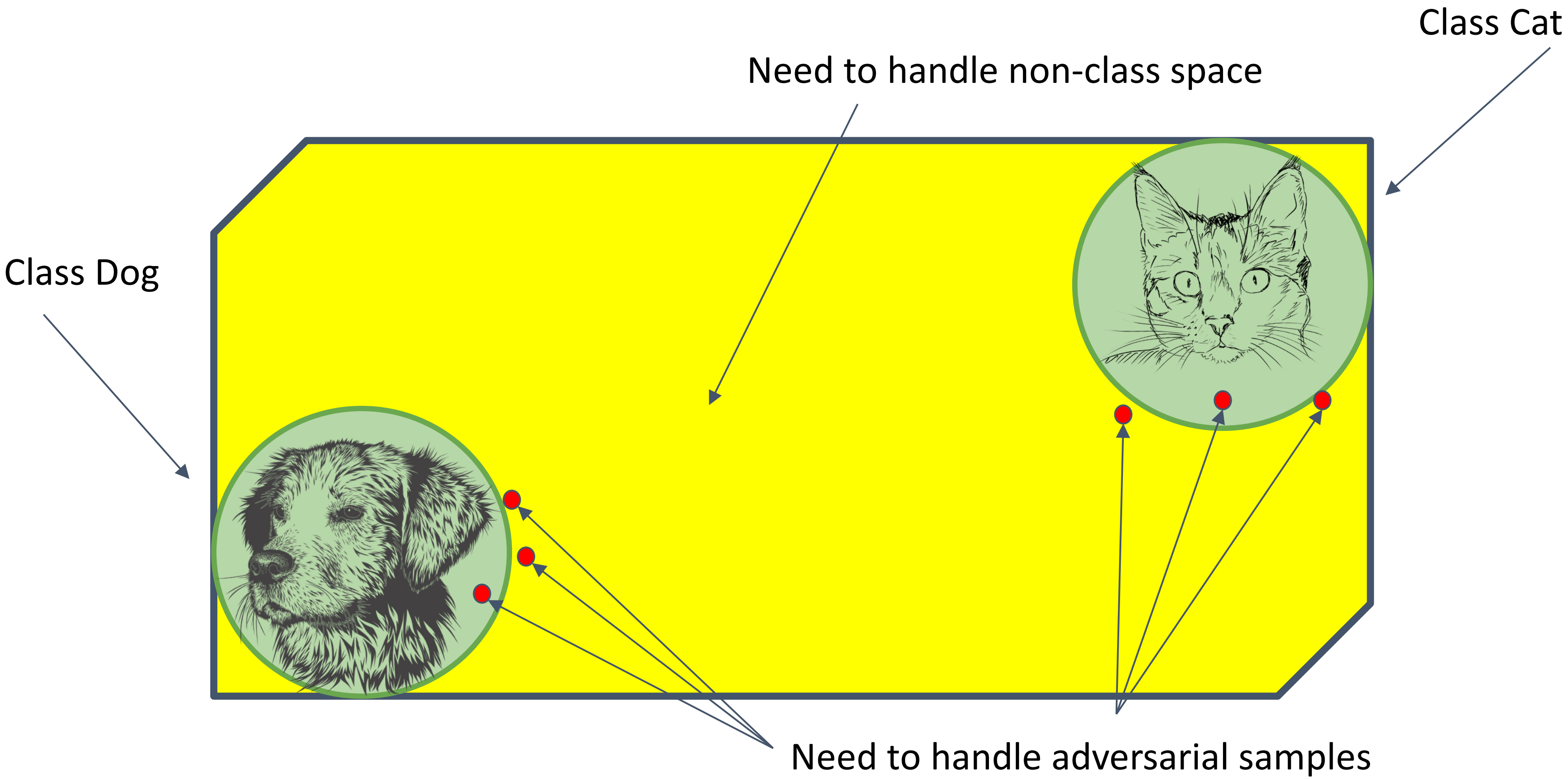
- Our sustainability work led to sponsorship from Bosch to look at machine vision
- Deep neural networks are much better at this but vulnerable to adversarial examples
- But are you really worried that someone will cause a car crash using a data projector?
- The right response may be fragility rather than robustness, so you get to know that you are under attack. How might we do that?

Adversarial inputs


Class: bird
Confidence: 0.9659422039985657
+
Difference
=
Class: automobile
Confidence: 0.8248467445373535

- From bird to car with a few tiny tweaks!
- Adversarial examples exist for all DNN models
- Attacks are findable and often transferable

Attack Detection



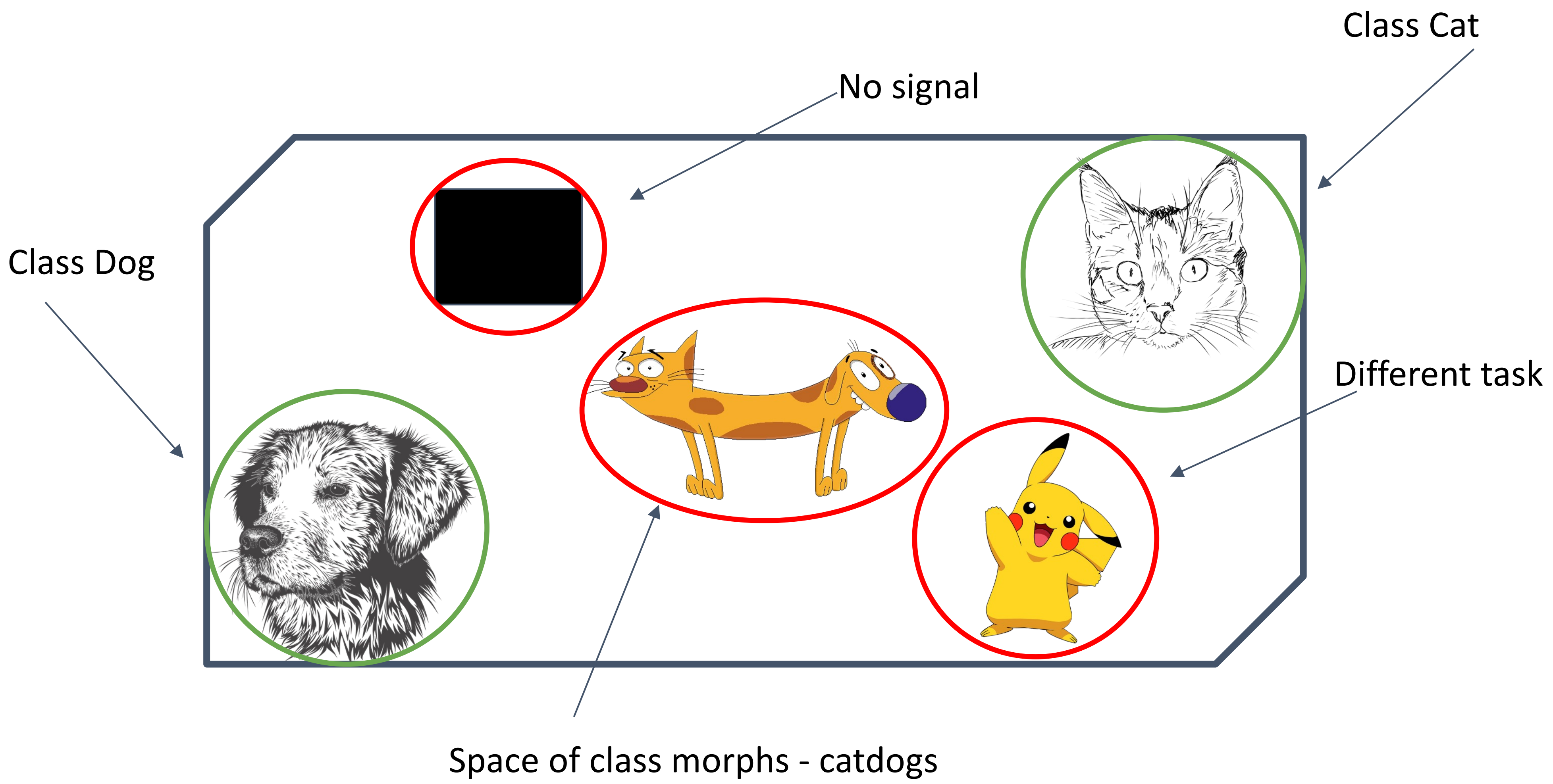
Class Cat

Need to handle non-class space

Class Dog

Need to handle adversarial samples

Attack Detection (2)



Class Cat

No signal

Class Dog

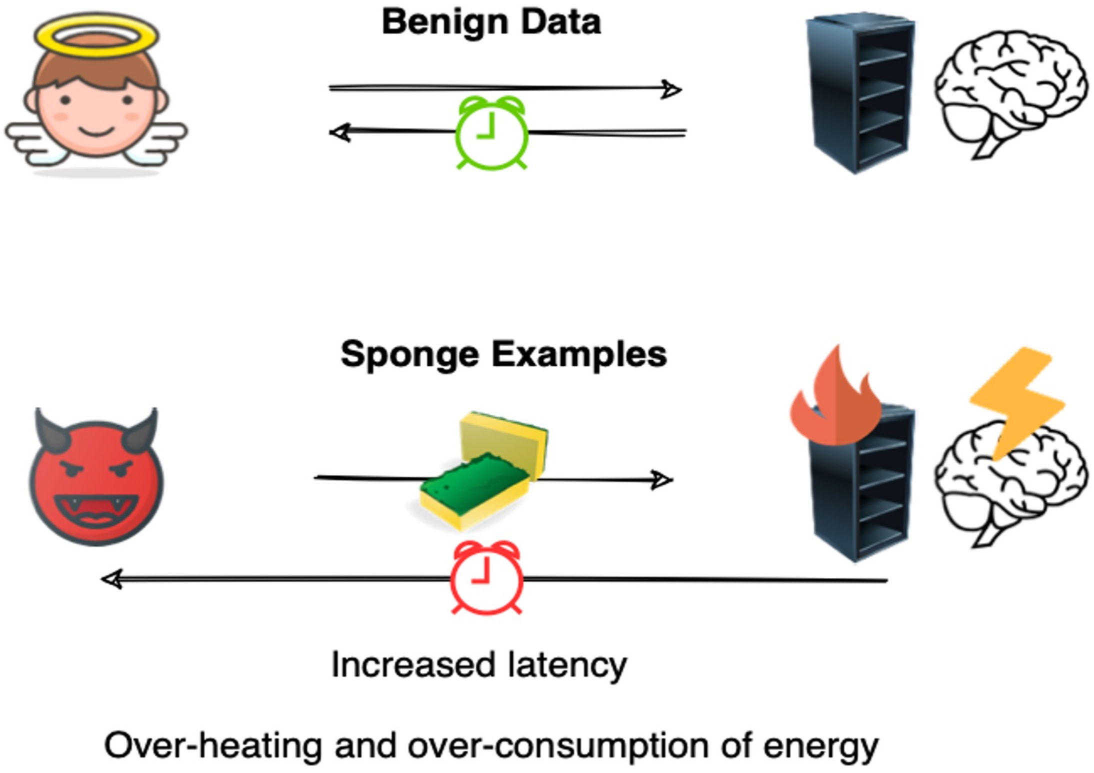
Different task

Space of class morphs - catdogs

Idea: the Taboo Trap

- You train your kids to have beautiful manners
- Then they go off to school and within a week know some words your mother doesn’t like!
- Breaking taboos => exposure to adversarial input!
- Can we set taboos (on outputs or activations) during training, and alarm when we see them?
- Answer: yes, this works rather well.
- Can diversify with different taboos – like crypto keys! (first interaction of crypto with ML…)

Sponge attacks



- *'Sponge Examples: Energy-Latency Attacks on Neural Networks',* I Shumailov, YR Zhao, D Bates, N Papernot, R Mullins, R Anderson, arXiv:2006.03463

Sponge attacks (2)

- We discovered a very wide range of sponge attacks, on all the hardware /algo optimization
- NLP systems are particularly vulnerable! You can use double meanings (the ‘conundrum attack’), or just drop a few Chinese characters in Russian text to stall a translation
- So ML systems must be designed for worst-case rather than average-case, or place limits on computation
- Real system engineers have known this stuff for decades, but today’s ML enthusiasts ignore it!

Bad characters

- Inspired by the discovery that Chinese characters hosed a Russian – English translation, my student Nicholas Boucher looked more carefully
- Unicode games were used in the early days of phishing to obscure URLs
- What sort of games can be played with machine translation systems?
- Plenty, it turns out!
- *Bad Characters: Imperceptible NLP Attacks, N Boucher, I Shumailov, R Anderson, N Papernot arXiv:2106.09898*

Homoglyphs

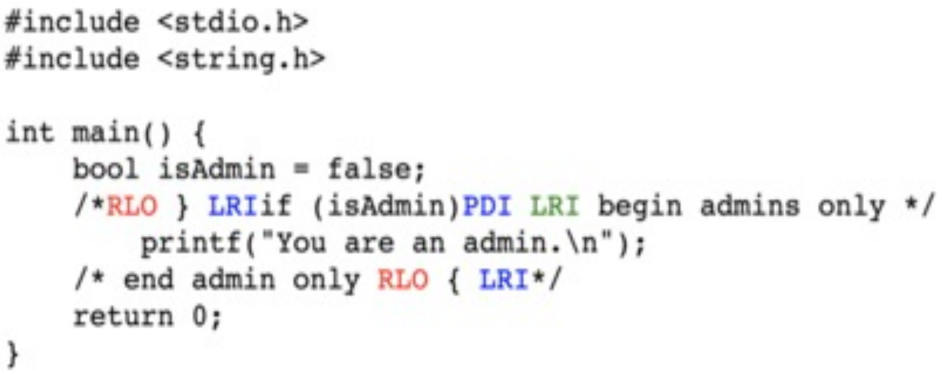
- Example: the normal ‘a’ and the Cyrillic ‘a’ render as the same glyph, but are different in Unicode
- You can often sabotage translation by swapping a handful of characters for homoglyphs
- You can often get a similar effect by dropping in a few zero-width spaces (yes, Unicode has them)
- This sabotages not just translation, but toxic content filtering
- Many potential abuse cases…

Even more devious…

- Unicode also has directionality control characters, which let you swap text between left-to-right and right-to-left
- E.g. to embed an English phrase in an Arabic newspaper
- So: we can write an email in English saying “please pay $1000 to account 123”
- Google Translates it to Spanish as “to account 321”
- MS / G / IBM should know to sanitise all inputs… !

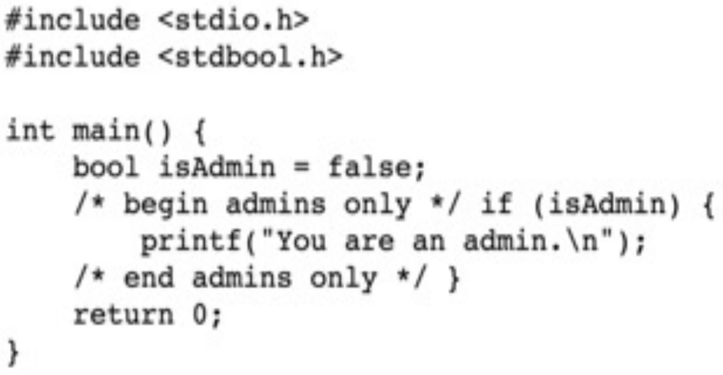
The Trojan Source attack

- It works on source code too!
- You can embed bidirectionaly control characters in source code, which compilers ignore if they’re in string literals or comments
- Result: the compiler sees one logic, and the human reviewer another



```
#include <stdio.h>
#include <string.h>

int main() {
    bool isAdmin = false;
    /*RLO } LRIif (isAdmin)PDI LRI begin admins only */
        printf("You are an admin.\n");
    /* end admin only RLO { LRI*/
    return 0;
}
```

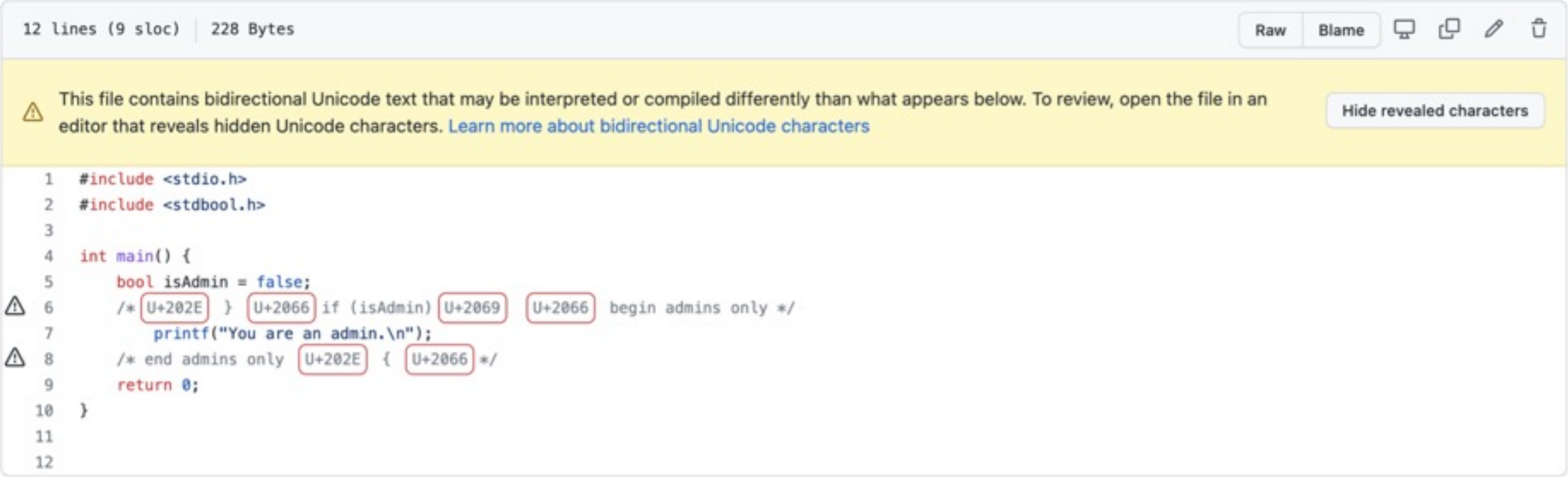


```
#include <stdio.h>
#include <stdbool.h>

int main() {
    bool isAdmin = false;
    /* begin admins only */ if (isAdmin) {
        printf("You are an admin.\n");
    /* end admins only */ }
    return 0;
}
```

Preventing the attack

- After we responsibly disclosed the Trojan Source attack to the major languages and code editors as CVE 2021-42574 and 2021-42694, many fixed it



⚠ This file contains bidirectional Unicode text that may be interpreted or compiled differently than what appears below. To review, open the file in an editor that reveals hidden Unicode characters. Learn more about bidirectional Unicode characters

```
#include <stdio.h>
#include <stdbool.h>

int main() {
    bool isAdmin = false;
    /*U+202E } U+2066if (isAdmin)U+2069 U+2066 begin admins only */
        printf("You are an admin.\n");
    /* end admins only U+202E { U+2066*/
    return 0;
}


```

Fixing code vs fixing ML models

- Most languages and editors fixed the bug eventually (Rust was keenest; Oracle/Java refused)
- Those who’d subcontracted bug reporting were harder; we had to get past the subcontractor
- However of the big NLP models on which firms increasingly rely, only Google did anything
- Firms relying on third-party NLP services for translation, hate speech detection and general UX tasks remain vulnerable

So what's going on?

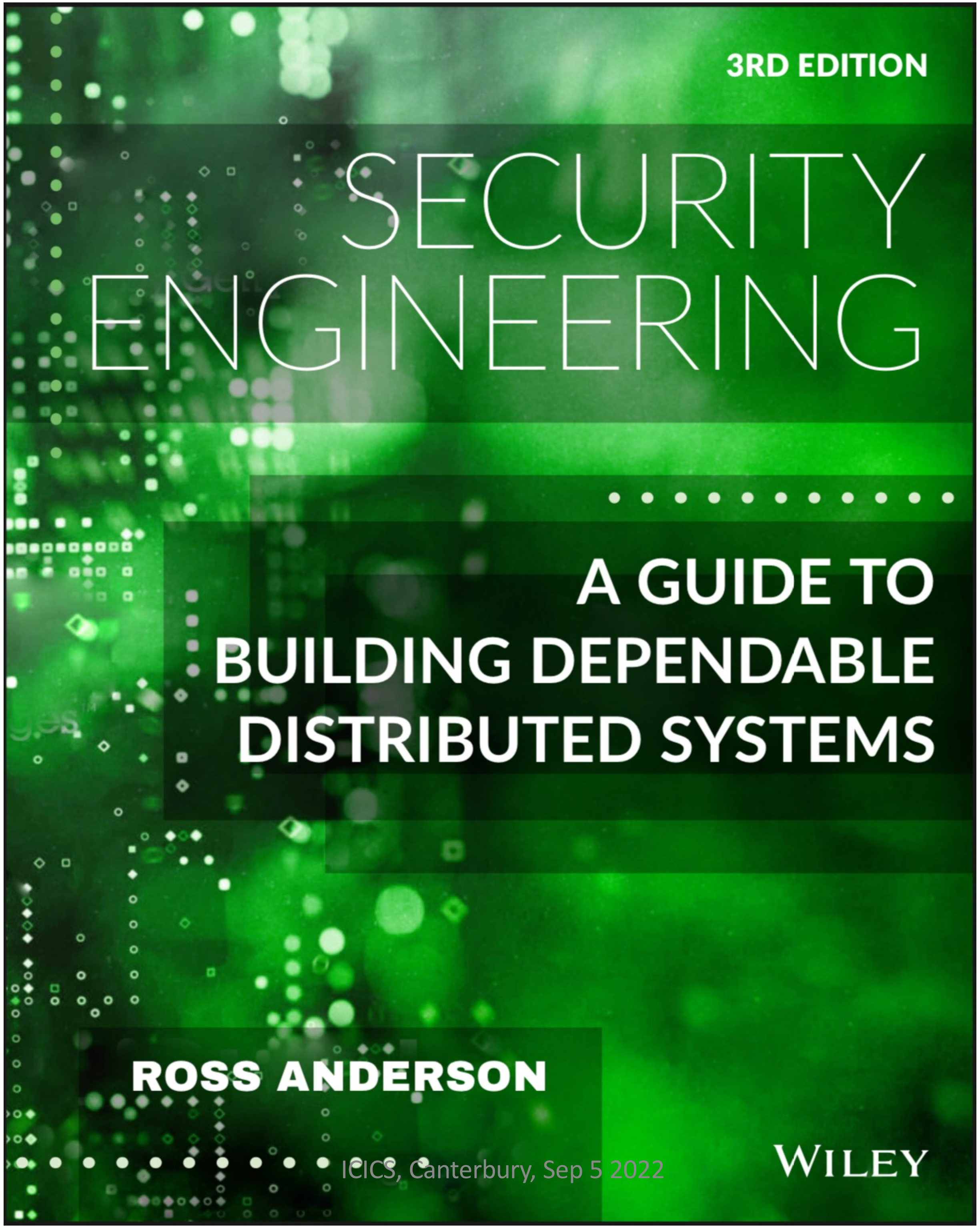
- Maybe ML models are too expensive to update? But you can sanitise the inputs easily enough
- Do ML vendors not know they need to do this? Surely not if they're IBM, MS, Google...
- At one, ML / security teams blamed each other
- Security, and safety, are whole-system properties!
- Other ML teams also tend to ignore this...

Topics for research on code vs ML

- Cost of an upgrade / bugfix
- Time to do an upgrade / bugfix
- Culture of C coders versus data scientists
- Expectations of dependability
- Publicity for code bugs versus ML misbehaviour
- Competition / market power
- Maturity of technology and market

New directions…

- Maintenance will be ever more of the cost of systems as they get more complex, and start to incorporate machine-learning components
- 30-year patching requires a more stable and powerful toolchain
- But ML may disrupt this!
- Do you engineer safety/security in the ML model, at the API, or end-to-end?
- And how can we motivate ML teams to patch?



3RD EDITION

SECURITY ENGINEERING

A GUIDE TO BUILDING DEPENDABLE DISTRIBUTED SYSTEMS

ROSS ANDERSON

WILEY