Software and Security Engineering

Lecture 2

Richard Mortier

rmm1002@cam.ac.uk

With many thanks to Ross Anderson and Alastair Beresford

ANDY GREENBERG SECURITY JUL 21, 2015 6:00 AM

Hackers Remotely Kill a Jeep on the Highway—With Me in It

I was driving 70 mph on the edge of downtown St. Louis when the exploit began to take hold.

https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/

2

[Watch first 3 minutes of the video. Ask the audience to write down all the aspects of the car which the remote attackers could control.]

Further reading and video: https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/

Architecture matters

- Lots of legacy protocols trust all network nodes
- Chrysler Jeep recall
- Defence in depth: separate subnets, capable firewalls,

3

There are a wide variety of legacy protocols in existence: DNP3 in control systems, CAN bus in cars, and so on. Many had either no stated security policy at all, or a security policy which does not take remote networking into account (e.g. the Internet).

As we just saw in the video, the Jeep Cherokee could be controlled remotely over the mobile network. This is a poor architecture: you don't want to allow unfettered remote control of the CAN bus, but equally well you need to for other things to function. How do you fix? Can we apply some of the ideas from models we saw in the last lecture (e.g. multilevel and multilateral security)? Defence in depth is also important in order to ensure the failure of one component does not lead to an accident.

Swiss Cheese Model

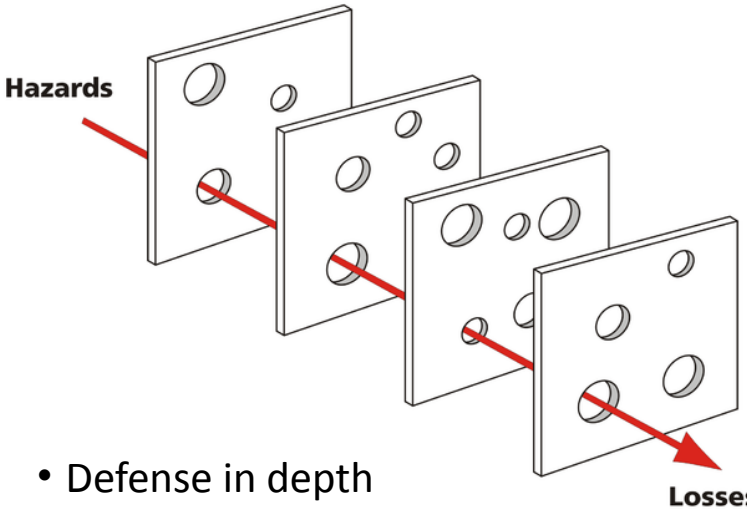


Diagram by
Davidmack
CC-BY-SA 3.0

- Defense in depth
- Layers could include hardware, software, policy, human factors, etc.

4

The Swiss Cheese model of accident causation is a model used in risk analysis and risk management. The aim is to ensure *defense in depth*: what might be open at one layer is closed at another. Defense in depth works provided that there is at least one layer which does not have a flaw which allows a hazard to turn into an accident. An accident occurs when the weakness in every layer lines up – something we wish to avoid.

https://en.wikipedia.org/wiki/Swiss_cheese_model

Safety policies

- Industries have their own standards and cultures, often with architectural assumptions embedded in component design
- Plethora of safety legislation
- Sometimes brand new standards, but in more mature industries safety standards tend to evolve
- Two basic ways to evolve:
  - *failure modes and effects analysis*
  - *fault tree analysis*

5

Safety is sectoral: those working on car safety don't talk to the aircraft industry.

Many industries are much more tightly regulated than the computer industry. For example, there are over 180 regulations for cars. Example: “ABS failure mustn’t cause asymmetric braking”.

Understanding and improving our understanding of the relationships between failures and outcomes can be bottom-up (failure modes and effects analysis) or top-down (fault tree analysis). Both approaches are useful.

Failure modes and effects analysis (bottom-up)

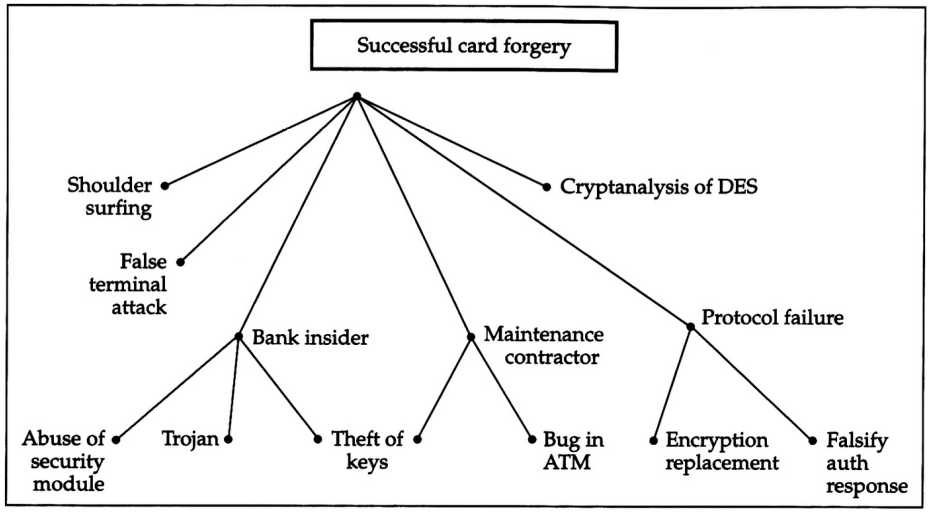
- Look at each component and list failure modes
- Figure out what to do about each failure
  - Reduce risk by overdesign?
  - Redundancy?
  - …
- Use secondary mechanisms to deal with interactions
- Developed by NASA

6

Example: You've got a person sitting on a rocket. What could go wrong? What would cause the rocket to explode on the ground? What would cause the second stage not to separate from the first? And so on. You can write down all the failures bottom-up. An extreme example: what would happen if bolt number 40674 fails when attached to a rocket fin? Could we reduce the risk of failure by increasing it's diameter, length or material used in manufacture? Could we introduce two bolts (redundancy) in order to reduce the risk of failure?

Example 2: For planes, the basic failure you worry about is the engine failing; examples include running out of fuel, engine on fire, etc. This is only a real problem when flying over mountains, forests or the ocean since otherwise an emergency landing is possible. The question you might ask is how long does the aircraft need to survive on the flight?

Fault tree analysis (top-down)



Work backwards from the bad outcome we must avoid to identify the critical components

7

This is the opposite to failure modes idea: Start at the top with a list of bad things you wish to avoid, and for each one work down.

[Talk through the examples on each of the nodes in the tree.]

This works particularly well if there's a small, finite, number of bad things which could happen: you start with each bad outcome and work your way down from it.

Example: nuclear bomb safety

- Don’t want Armageddon caused by a rogue pilot, a stolen bomb, or a mad president, so require

- Authorisation: president releases code
- Intent: pilot puts key in bomb release
- Environment: *N* seconds zero gravity

- Independent, simple, technical mechanisms

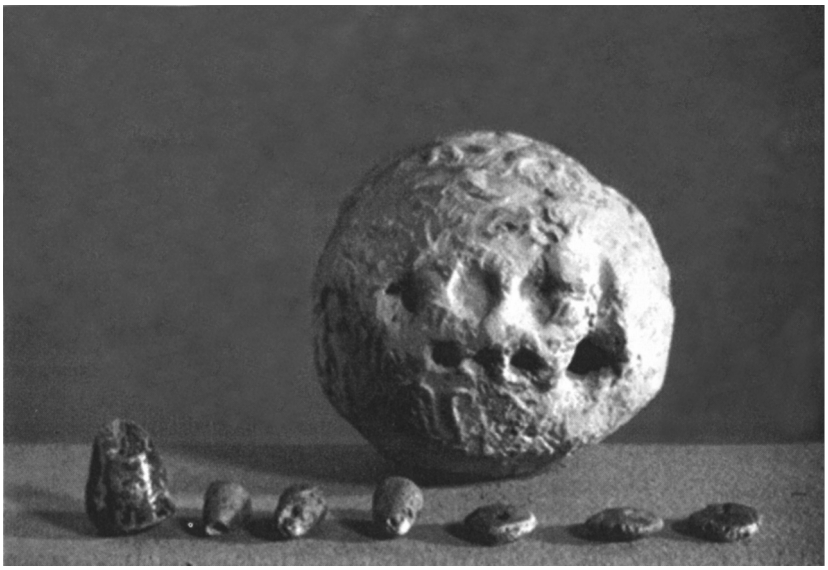
8

NATO countries require three things: authorisation, intent, environment.

- Authorisation: there's a code which needs typing into the weapon. This code is kept by a responsible person with authority to use the bomb.
- Intent: the pilot needs to decide when and where the bomb will be released.
- Environment: This is really important since it's quite hard for the malicious person to get, say, 20 seconds of zero-gravity without access to a jet engine.

We hope that these things are orthogonal and therefore you can only set off a bomb unless you have the bomb, a mad or otherwise compromised president, a well-trained evil pilot, and access to a jet plane.

Bookkeeping, 8—4th millennium BCE



9

These are bullae from the British Museum. Each bulla records the stock stored in the granary. When you deposit the wheat or olive oil you receive a bulla back; you can then present your bulla later in the year to get the goods back. Note that the bulla were pushed into clay, and the clay is then fired so that you had a seal (each party kept one half and therefore this lets each other validate the outcome.)

This is where writing comes from: “As the clay tokens and bulla became difficult to store and handle, impressing the tokens on clay tablets became increasingly popular. Clay tablets were easier to store, neater to write on, and less likely to be lost. Impressing the tokens on clay tablets was more efficient but using a stylus to inscribe the impression on the clay tablet was shown to be even more efficient and much faster for the scribes. Around 3100 BCE signs expressing numerical value began. At this point, clay tokens became obsolete, a thing of the past.” https://en.wikipedia.org/wiki/Bulla_(seal)

Bookkeeping, circa 1100 AD

- Double-entry bookkeeping: each entry in one ledger is matched by opposite entries in another
- Ensure each ledger is maintained by a different subject so bookkeepers must collude to defraud
- Example: a firm sells £100 of goods on credit, so credit the sales account, debit the receivables account. Customer subsequently pays, so credit the receivables account, debit the cash account.

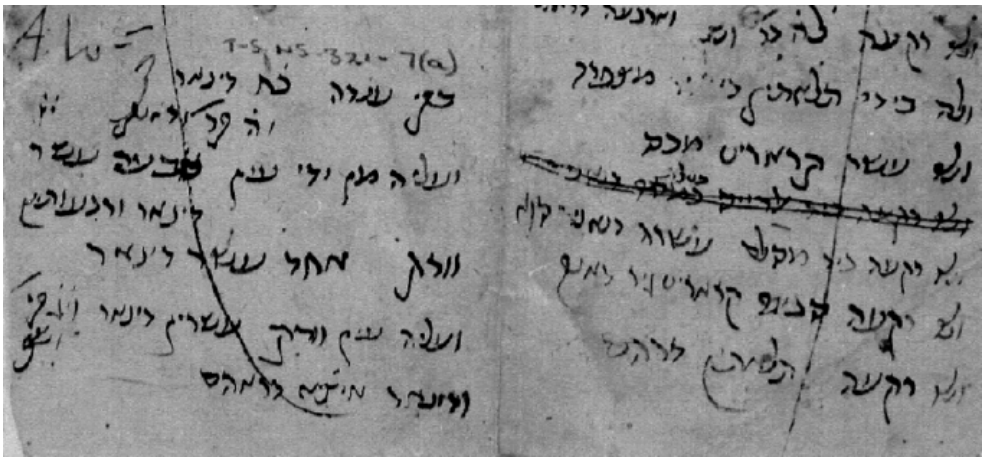
10

How do you manage a business that's grown too big to staff with your own family members?

Double-entry bookkeeping allows you to check the behaviour of bookkeepers are consistent. This leads to the phrase "the books balance". It requires separation of duties: in other words, that subjects cannot take on two roles in the bookkeeping system such that a single subject can commit fraud and ensure that the books still balance.

Further information, including the ability to build your own simple accounting system, can be found here: https://anvil.works/blog/double-entry-accounting-for-engineers

Double-entry bookkeeping found in the Genizah Collection



11

"Jewish bankers in Old Cairo used a double-entry bookkeeping system which predated any known usage of such a form in Italy, and whose records remain from the 11th century AD, found amongst the Cairo Geniza"

https://en.wikipedia.org/wiki/Cairo_Geniza

Separation of duties in practice

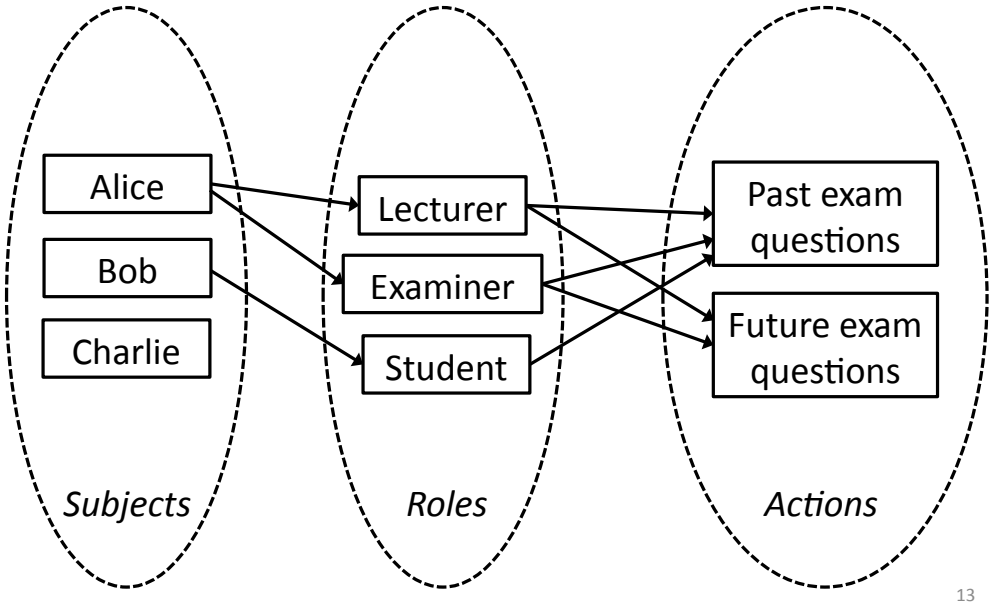
- Serial:
  - Lecturer gets money from EPSRC, charity, …
  - Lecturer gets Old Schools to register supplier
  - Gets stores to sign order form and send to supplier
  - Stores receives goods; Accounts gets invoice
  - Accounts checks delivery and tell Old Schools to pay
  - Lecturer gets statement of money left on grant
  - Audit by grant giver, university, …
- Parallel: authorization from two distinct subjects

12

Serial or sequential separation assigns a different subject to each role in the double-entry bookkeeping model. In the example, the lecturer interacts with many distinct employees of the University in order to carry out financial transactions. These provide checks against malice or mischance.

An alternative to the sequential approach is the parallel model which requires two subjects to sign an agreement. This might be used for large, irreversible, operations such as signing a guaranteed cheque, or signing a significant contract.

Role-Based Access Control (RBAC) decouples policy and mechanism



RBAC adds an extra level of indirection between the subjects and the actions a subject can carry out. We call these intermediate states *roles*. Roles provide flexibility. For example, in a university with 20,000 people, you can't set up individual accounts with separate Unix file permissions for each person. Instead, you define roles and then assign roles to people. This way the complexity is manageable. Example roles in other domains might be Officer of the Watch in the Army, Branch Manager in a banking context, Charge Nurse at a hospital, and so on.

RBAC helps with complexity, but doesn’t remove it entirely. You still need to write the policy, and this policy might be wrong.

It is also possible to combine RBAC with other models. For example, SELinux offers MLS with RBAC.

Summary of security and safety

- What are we trying to do?
- Security: threat model, security policy
- Safety: hazard analysis, safety standard
- Refine to protection profile, safety case
- Typical mechanisms: usability engineering, firewalls, protocols, access controls, …

14

First, define what you are trying to do at a high level. Some examples:

1. Stop a story reaching the front page of The Guardian
2. Prevent a Branch Manager from running off with the cash
3. Etc.

Given an overarching aim, then construct a protection profile or high-level safety case. What are the threats? Look at the failure modes (bottom-up) or conduct a fault-tree analysis (top-down). This will allow you to construct a detailed security target or safety case, which will inevitably involve consideration of specific *mechanisms*.

We will now look at different mechanisms, starting with user behaviour.

Do not ignore user behaviour

- Many systems fail because users make mistakes
- Banks routinely tell victims of fraud “our systems are secure so it must be your fault”
- Most car crashes are user error; yet we now build cars with crumple zones

15

This is the most overlooked mechanism. Serious accidents can occur by ignoring the limitations of humans…

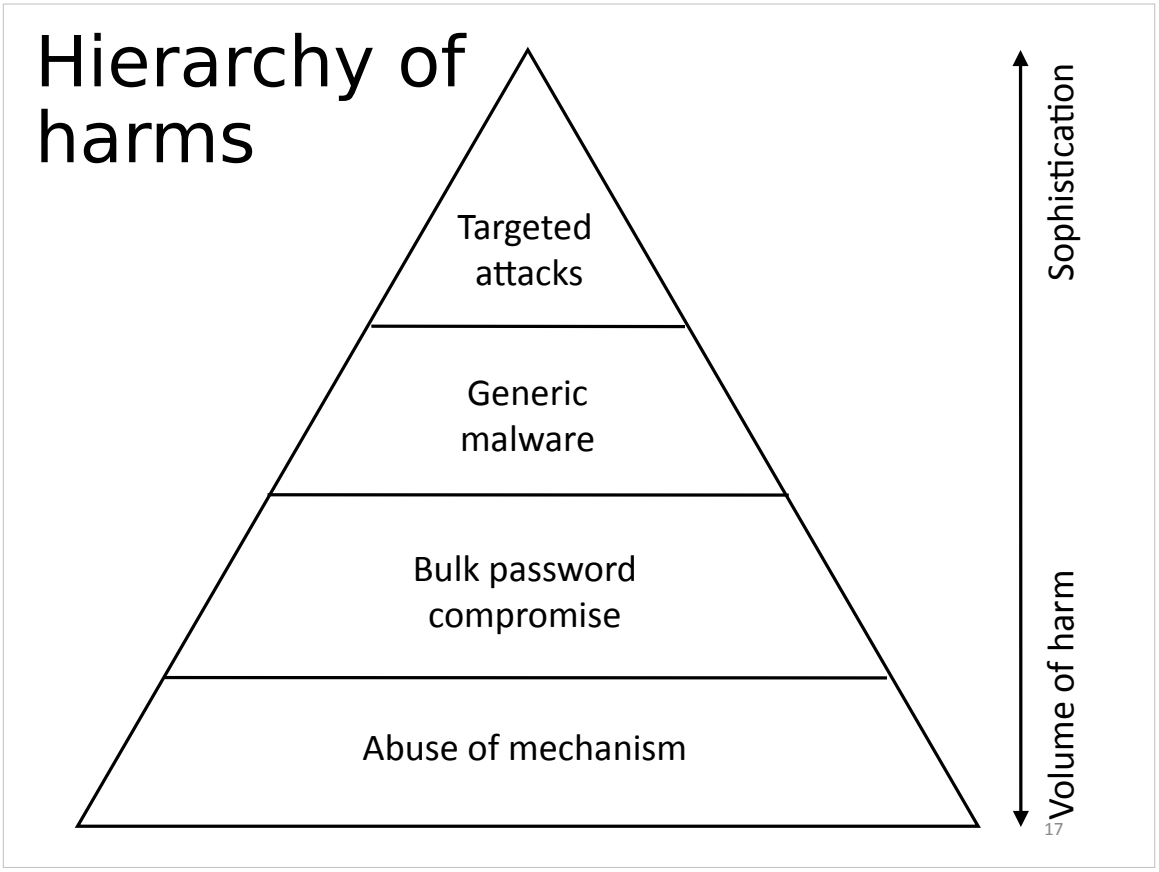
Example: Tell the user to choose a password that can't be guessed and don't write it down. Then ask them to create a different password for each of the hundreds of websites that they use (some infrequently). Then say you must change each of these passwords every three months. This is simply not an acceptable cognitive load on an individual, so coping mechanisms arise. Can you think of specific examples of poor understanding of user behaviour in computer systems? What are your coping mechanisms?

Chevrolet 1959 vs 2009

https://www.youtube.com/watch?v=fPF4fBGNK0U

16

For many years the car industry prospered while blaming the user: “it's not our car which is defective, but the driver…”. The mantra from the manufacturers was “sue the driver not the manufacturer”. It wasn't until the 1960s that the users managed to challenge this state-of-affairs. It took many decades for the motor industry to get to this point. This sort of dumping of failures onto the user happens time and again as new industries start, and the computer industry is no exception.



Cybercrime today makes up around half of all crime by both cost and volume.

The hierarchy of harms: targeted attacks, generic malware, bulk password compromise, abuse of mechanism. With each step down in this hierarchy, the number of victims goes up by an order of magnitude or more. Most harm occurs at the bottom, and perhaps not too surprisingly these attacks are also the least technically sophisticated.

At the top are the targeted attacks (e.g. Russian intelligence get access to Hilary Clinton's email), but these are really rare. Bulk malware, such as Zeus or Dridex, infects millions of users. Bulk password compromise for 50-100 million people. Perhaps cracking the password file and guessing that the users use the same passwords for Gmail and provides access to 100,000 email accounts.

Many abuses of mechanism

- Cyberbullying
- Doxxing
- Fake rental apartments
- …

How can we protect against these attacks?

18

Cyberbullying. Example: using existing messaging platforms to bully others

Doxing. Researching then broadcasting private information of the victim.

Fake rental apartments. We have seen websites advertise apartments in Cambridge which either don't exist or rather they aren't for rent and then you cheat out of the deposit. What can the website do to push back on this? What can the University do? Well, the University could write to all accepted applicants and tell them to use the official accommodation service and warn them of the scams... but this still doesn't work. Indeed we need the ideas -- this is very much an unsolved problem.

Useable privacy is also hard

- Traditional approaches – anonymisation and consent – are really hard to deliver
- Problem gets harder as systems get larger
- Automated data collection (e.g. from sensors) makes the situation more difficult again

19

Hierarchy of harms is often focused on security or safety. Privacy also needs to be usable.

The traditional solutions: informed consent and anonymisation.

Consent is hard to do right – how do you know users made informed decisions? Is there really genuine choice on current platforms with their often impenetrable privacy notices?

Anonymisation aims to remove personally-identifying information from a dataset while still preserving utility. Unfortunately this is extremely difficult. Consider, for example, location data. It turns out that where you live and where you work is often unique, so ”anonymous” traces of the movements of people can be reidentified simply by combining the location trace data with the electoral roll (home location) and employee database (where you work).

Further reading: Golle, Philippe, and Kurt Partridge. "On the anonymity of home/work location pairs." International Conference on Pervasive Computing. Springer, Berlin, Heidelberg, 2009. https://link.springer.com/chapter/10.1007/978-3-642-01516-8_26

How many CPUs in the pictures and how does this system differ from the car? [Ask audience] This is the intensive care ward in Swansea. [Click next to run animation]

Note the difference from the car: In a car they are all on the CAN bus and integrated; in the hospital they are all separate with their own interface.

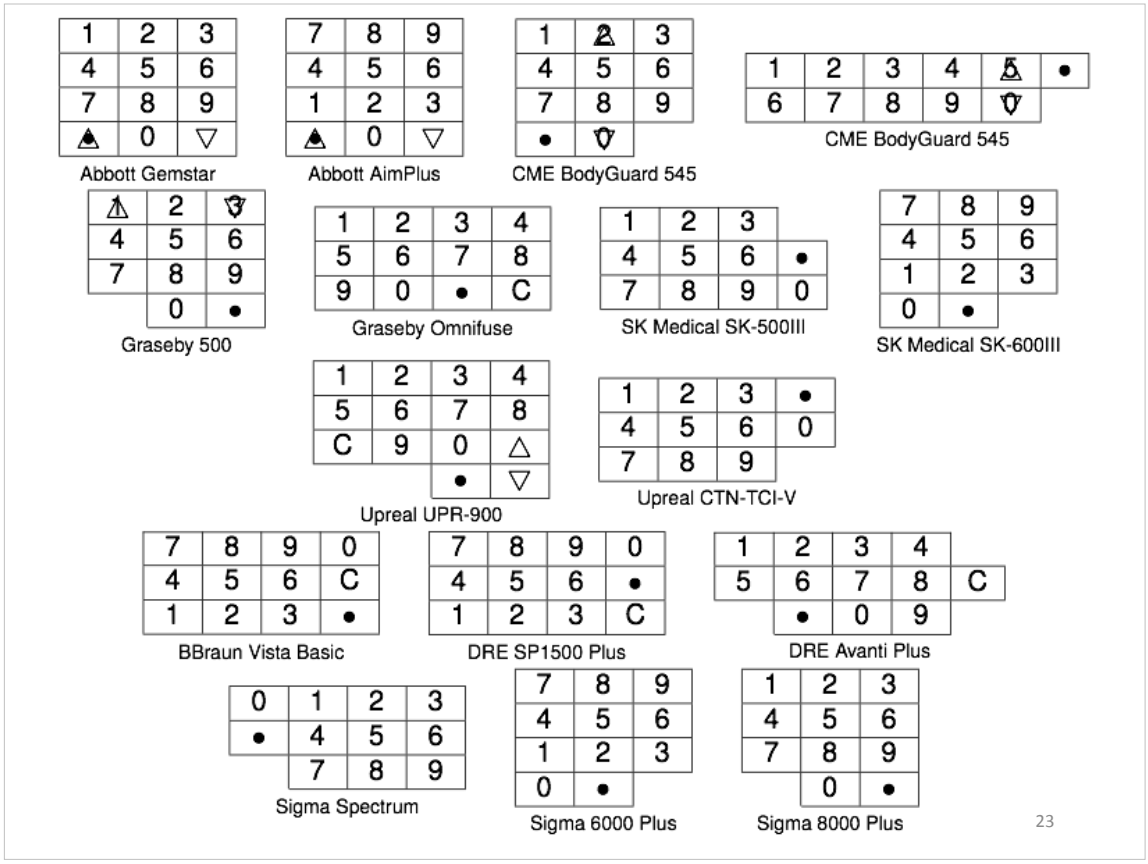
Here are seven infusion pumps. Note that all the controls are different!

“Approximately 11% of patients in UK hospitals suffer adverse events, of these half are preventable, and about a third lead to moderate or greater disability or death [1]. Medication errors seem to be one of the most preventable forms of error: more than 17% of medication errors involve miscalculation of doses, incorrect expression of units or incorrect administration rates [2]. The Institute of Healthcare Improvement’s “global trigger tool” suggests adverse events may be ten times higher [3]. These figures come from research in different countries with different methodologies and assumptions, and suffer from a lack of reliable information [4], but there is general agree- ment that preventable mortality is numerically comparable to road accident fatality rates [5]."

Quote from: Thimbleby, Harold. "Improving safety in medical devices and systems." 2013 IEEE International Conference on Healthcare Informatics. IEEE, 2013. https://ieeexplore.ieee.org/abstract/document/6680455

How do we standardise? Even where there are standards, they aren't followed: International standards say that litres should be written with a capital-L, but note that some don't do this (and therefore a lowercase-l could be confused with a one).

Even the same device make and model (here BodyGuard 545) have different versions with different user interfaces.



Here are a range of keyboard layouts from infusion pumps. They are all different. If the layout problems weren't enough, there are also challenges in how key presses are interpreted.

Example problem in this area, although from the banking domain: "In 2008, Grete Fossbakk transferred 500,000 kroner to her daughter using the web interface to her Union Bank of Northern Norway account. Unfortunately, she admits, she miss-keyed her daughter's bank account number and a repeated 5 in the middle of the account number made it too long. The Union Bank of Northern Norway's web site then silently truncated the erroneous number, and this new number (which was not the number Fossbakk had keyed) happened to match an existing account number. The person who received the unexpected 500,000 kr spent it. Only on the steps to the court room did the bank relent and refund Fossbakk."

(quote from Thimbleby's paper). See also K. A. Olsen, "The $100,000 keying error," IEEE Computer, vol. 41, no. 4, pp. 108–106, 2008.

Similar problems occur with pocket calculators, which are also used by nurses to calculate medical doses.

Medical device safety

- Usability problems with medical devices kill about the same number of people as cars do
- Biggest killer nowadays: infusion pumps
- Nurses typically get blamed, not vendors
- Avionics are safer, as incentives are more concentrated
- Read Harold Thimbleby's paper!

Bulk password compromise

- Example: in June 2012, 6.5m LinkedIn passwords stolen, cracked (encryption did not have a salt) and posted on a Russian forum
  - Method: SQL injection (see later)
  - Passwords were reused on other sites, from mail services to PayPal.
  - Reused passwords were used on those third-party sites
- There have been many, many such exploits!
- What can we do about password reuse?

25

[Ask the audience to chat to their neighbour and write their own top-5 list of things they would do to prevent password reuse. Discuss.]

Phishing and social engineering

- Card thieves call victims to ask for PINs
- A well-crafted email sent to company staff, with apparently authority, can get 30% yield
- Some big consequences (see next)
- Think like a crook (see Mitnick reading)

26

Phishing or social engineering aids an attacker at all levels in the hierarchy of harms. An example story. A malicious person finds a card in the street and phones up the owner and says "Hi it's Barclay's here... We have noticed your card was used incorrectly in a number of transactions, have you still got your card?". "I'm sorry, I lost it in Tesco 30 minutes ago, I was going to ring you!". "No problem, we can you sort it out for you, could you just tell us your PIN so we can cancel your card for you?". Sending a generic email to all employees, perhaps purporting to be from the boss, with a URL in it can be very effective. The email simply needs a cover story. This could be anything from “please complete the staff survey” to “your inbox is full; please click here to increase your quota”.

Another example story, this time spearphishing. An attacker compromises an email server and finds the inbox from the CEO. The attacker then examines the recent email traffic, and works out who the financial controller (chief clark) is and also what recent business is going on which might lead to transfers of cash out of the company. The attacker then crafts an email purportedly from the CEO to the financial controller asking him or her to send a large sum of money to a plausible sounding company whose bank details are actually under the control of the company.

The Chinese Government wanted to infiltrate the Dali Lama’s office and ended up compromising 30 out of the 50 or so computers. The way in appears to have been a compromise of one Monk's computer. This machine was used to compromise the mail server used by the Monks, so that when an attachment is sent from one Monk to another it could be rewritten to include malware in the attachment and therefore spread the attack to more machines . This is great for covert ops – all mail sent and received is legitimate, so the Monks can check with each other to see if they did send the email (which they did) but it's still malicious!