The Real World Intrudes

I.G.Batten@bham.ac.uk

Green Field Sites

- **Scope** the ISMS
- Build **Asset Register**
- Analyse **Threats**
- Build **Risk Register**
- Impose **controls** to control risks
- **Operate** and **measure**
- **Improve**

Plus: **respond** to unexpected events (incidents) and **learn** from them

But…

- It’s rare that you build an ISMS from scratch at the same time as building an IT infrastructure from scratch.
- And even if you were doing this, you probably would not have a clear enough understanding of the landscape to get either right first time.
- Normally, we are retrofitting an ISMS to existing infrastructure.

Case #1: Finance

- Strong security culture
  - although more about shotguns, internal fraud and customer fraud than information risk.
  - The IT might not be part of it, but there is staff culture to build on.
  - And there is an internal control and audit function and culture you can leverage.
  - Security is seen as core, or core-ish, and therefore gets management attention and investment.
  - Although IT is often semi-detached and/or outsourced.

Case #2: UK Healthcare

- Worryingly lax attitude to privacy, for deep cultural reasons outside the scope of this course
  - Low priority, until it goes wrong, when it's a mad panic
- Complex, "evolved" IT environment with a lot of interacting systems which were procured in isolation, a lot homegrown, a lot on obsolete platforms.
- Minimal audit, and almost entirely related to medical outcomes and costs (NHS "Counter Fraud" don't really do IT).

Case #3: SME

- For example, solicitors and builders subject to invoicing fraud.
- Low priority as compared to paying the bills and keeping customers happy
- “Nothing we have is worth stealing”
- IT is mostly bought-in, and operated by people who don’t know and don’t care much more than how to do their daily tasks.

Common Problems

- Management focus
- Legacy
- Staff training and attitudes

ISMS on a Brown Field

- Building on brown field sites starts with remediation: removing the toxic waste from the past.
- You will almost certainly have things in the infrastructure you take over which are giant waving red flags for an ISMS.
- But how do you manage the change?

Side track

- Just going to run over a few management fallacies and problems which make our lives in IT much harder than they need to be.
  - Technical Debt
  - Sunk Capital Fallacy
  - Stranded Capital Fallacy
  - Reluctance to Rent
  - Reluctance to Change

Technical Debt

- It’s cheaper, in immediate cash-flow terms, to keep paying the interest on your credit card than it is to pay off the capital debt. Even though it’s cheaper in the long run.
- Similarly, businesses accumulate **technical** debt, which they have to pay the “interest” on (ie, the cost of maintaining broken / wrong / expensive things) in preference to the larger, in the short term, cost of fixing it.

This year’s best example

- Apple’s on-disk filesystem, HFS, was designed in the early 1980s to support hard drives of a few tens of megabytes (and floppy disks!)
  - HFS+ conceptually the same, 1998, MacOS 8.1.
- Ported into Unix for OSX, and primary OSX filesystem until 10.12 (2016).
- Hideous: slow, poor crash recovery, **encryption a nasty bolt-on**, poor behaviour on SSDs.

Several failed attempts

- Port of Solaris's ZFS (itself a 2001–04 response to late 1970s filesystems) done but stalled for political / business reasons before being abandoned. ZFS supports native encryption, with lots of features of interest to security-conscious users.
- Multiple in-house projects failed
- Finally apfs rolling out in 10.13 starting last month (on everything from Watches to Pros).
- For older readers, see also the OSX transition

For security?

- Cheaper and easier to patch and mend rather than reimplement
- Quality lower and you are carrying the technical debt of not doing the job properly
- When building an ISMS, you would like to tear up everything and start again: effectively, **repay all the technical debt in one transaction**.
- Good luck with getting the resources for that!

Sunk Capital

- Suppose you have spent £1m on development costs for a project, and it has failed. Every year it costs £1m more than it brings in.
- Only the small amount of equipment you bought to do the projects with is saleable.
- What should you do?

Sunk Capital

- Rationally, you should give up and scrap the project, writing off the million pounds.
- You have spent the development costs whatever happens, but you don’t have to keep funding the losses.
- However, businesses aren’t rational, and a lot of messy excuses are often made to keep past failures going
    - Sometimes concern about the accounting implications of the write-off, forgetting the old business dictum that **Cash Is King** (see also “turnover is vanity, profit is sanity but cash is reality”).

For Security

- Often you want to move from an old system to a new one as part of imposing new controls
- But the old system cost a lot of money, so even if the new system is cheaper, it still involves writing off the old one.
    - Ironically, this is worse for companies which recognise the non-capital (time, effort) which went into the old system!

Stranded Capital

- Variation on Sunk Capital
- Suppose you have just bought a new car.
- It will cost you £30 to drive to London and park, but you can go by train for £10 (and you can, by the way, at the weekend).
- Which do you do?

Stranded Capital

- A lot of people would take the car, because otherwise it is “wasted” by being sat “unused”.
- Similarly businesses spend money to keep using things that can be done cheaper by (often) newer technologies.
- Famously, CS bought an interface card to keep using a disk drive on a new machine. Which cost **more** than replacing the disk drive with a larger, faster, more modern one.
- Again, fear of write-off, and irrational attitudes to “waste”.
    - A glance at my hi-fi reveals this tendency.

For security

- Often, a good security approach is outsourcing (risk transfer)
- Sunk and Stranded capital fallacies often make people reluctant to move from in-house solutions to outsource solutions
- Often a mask for sentimental attachment
- One of the reasons why incoming management can turn around businesses: much less sentiment.

Reluctance to Rent

- There is a very British thing about not wanting to pay rent. It is seen as “waste”.
- So the British are obsessed with buying holiday homes (hence why they are the #1 target for timeshare scams) even if it makes absolutely no financial sense.
    - They assume the capital is at least recoverable, ideally an investment in its own terms, and therefore over-estimate the cost of renting.
- Also fear that the rent might increase

Reluctance to Rent

- So if you can buy in credit card processing for 7%, many people will see that as 7% waste and pay irrational amounts of money to avoid that cost.
- As we know, a fully-compliant PCI-DSS solution (on which you still pay merchant fees to the card processor, by the way) is expensive and difficult.

For security

- Businesses of all sizes accumulate lots of small, in-house solutions which mask a lot of risk.
- Their costs are hidden in general IT overhead.
- Taking them out requires spending explicit money and often paying a regular monthly or percentage charge.

Reluctance to Change

- Taken together, these factors make people reluctant to change equipment and processes.
- There is some historical precedent, and I am caricaturing the position: change projects tend to **understate costs** and **overstate benefits** — cf. High Speed Two.
- But when rolling an ISMS out, you are often asking for a lot of change in a short period, and you need to think about how to manage this.

Existing Process

- Small companies often have no IT process at all. Kevin in IT deals with it all, and it's all in his head.
- But let's assume there is some process in place.
  - Is it being followed?
  - Is it fit for purpose?
  - Does it (or could it) generate records?
  - Could it be adapted?

Existing Process

- Some might address risks you have identified, or could be changed to do so.
- Some might address risks you don’t think are worth treating, or don’t even appear on your risk assessment.
- There is then a delicate balance: keeping the process is arguably stranded capital fallacy, but is the argument worth the saving?
- Choose your battles.

Existing Equipment

- Similarly, you will have equipment that was bought for purposes you don’t want to continue dealing with in-house.
  - For example, that very expensive PCI-DSS solution for card processing, when you’re pretty sure it would be better to go to WorldPay.
- You want to tell the business to accept paying 7%.
- Can you find something else to do with the equipment? That stops people fixating on the write-off.

Existing Staff

- This is where it gets very difficult
- If you are imposing a new ISMS, you may need new staff with new skills, and you may no longer need older skills (particularly in maintaining those old systems you plan to get rid of).
- The cost of change, and the willingness to change, will vary with company culture and legal position (Texas and Frankfurt have wildly different employment law!)
- In the UK, TUPE is a substantial issue.

Existing Staff

- Young dynamic management underestimate the value of institutional memory, particularly when there is change.
- And more experienced staff will be more trusted in the enterprise, so can front-up the changes you are making.
- Wholesale change for a new team is probably the wrong answer, but I am not a management consultant.

Starting the Process

- So we have an idea of the challenges we will confront, so what do we do next?