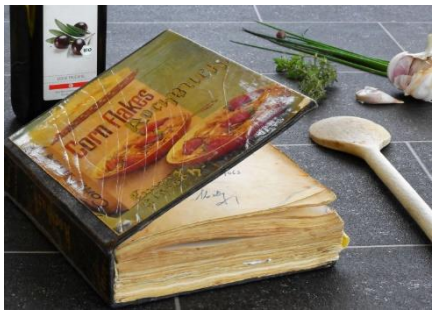
TBSE – a 'slightly technical' description

I am working on writing a technical paper describing TBSE in full detail that I will publish in a peer-reviewed journal due course. It is still in draft so I cannot make it available through my web site yet. In the meantime, I have written this short 'slightly technical' description of TBSE as an introduction for people. Here I describe the benefits of treating cyber security as a science, explain what that would look like and how TBSE enables us to do that today, and suggest three easy ways people could give it a try to see what it can do for them.

The way we think about cyber security needs to change



Cyber security is a highly technical and largely technological subject. That disguises the fact that we practise it as a craft, not a science. We have a series of 'recipes' (we call them Best Practices and international standards) that usually work well enough, but they have been compiled over time by bringing together common responses to attacks and breaches, not designed using scientific methods, data, analysis and results. That doesn't prevent us being able to build tolerably secured cyber systems but it does limit our ability to adapt and innovate.

It is as if we were Master Bakers in an age before the scientific understanding of chemistry. Consider the Master Baker's predicament. He has a number of recipes for making different types of bread, and he can reliably produce good loaves every day provided he is careful how he follows the recipes, always uses the same ingredients, and doesn't change ovens.

But has doesn't understand the basic chemistry going on as he makes each loaf of bread. He knows all recipes have to use sugar and salt, flour and water, and a small amount of that fragile magic ingredient called yeast, but he doesn't understand what is going on at a chemical level as the dough sits there rising. That limits his ability to innovate.

It means the only way he can come up with a better loaf of bread, or adapt his recipes to a different grain or oven, is trial and error. He could experiment to see what happens if he adds a bit more water to the dough. It makes the loaf sink. So he plays around adjusting the amount of water until he reliably gets good results. He could try using less salt to see if that lets him cut his costs by reducing the amount of sugar he needs, but he just ends up with a bland-tasting loaf nobody wants to buy. By changing the type of flour, he can see that some grains produce a nicely risen loaf while others always end up giving him flatbreads. Why is that?

If, instead, he had an understanding of the underlying chemistry that was going on, of how the yeast, sugar, water and gluten combine to give a soft light dough when mixed together, he could optimise the quantities of each ingredient, adapt his recipes for different grains and equipment, and save time and cost all in one go. No more failed loaves and no more wasted ingredients to explain away.

And that is how it is with cyber security. The Best Practices and international standards we use today provide us with

an uncertain but probably sufficient level of security provided we are careful how we follow them and provided we are not operating in an unusual threat environment. But we can't optimise those practices to suit our particular threat and technology situation or to maximise cost efficiency.

And there is more. At least when baking bread, we can tell if the recipe has failed straight away. We can see if the loaf doesn't look right, and one bite tells us if it tastes off. And if it has failed, we can have another go to see if it was the recipe or something we did that was wrong. All we will have wasted is a few affordable ingredients and a small amount of time.

We can't do that with our cyber security recipes. We don't have any way to measure the amount of security protection a given set of products and practices provides. Protection (or risk, if you prefer) is an intangible. We can't see it. We can't touch it. All we can do is apply a set of security measures and wait for something to fail. And when we get a security failure, do we try the same recipe again to see what went wrong? We don't want to take risks with our recipes because security failures can be disruptive and expensive. So we end up with a set of security recipes we cannot easily adjust, no way to measure how much protection we get from them, inconsistent results from one enterprise to the next, and no certainty as to why they fail when they do.

We have learned to live with these shortcomings. But it doesn't have to be this way. Look at how medical science has improved healthcare beyond anything doctors could have imagined half a century ago. Look at how materials science has enabled engineers to build bridges over huge expanses of water Brunel could never have spanned. I won't claim that treating cyber security as a science would save lives in the way medical science has, but it could certainly revolutionise the way we practise cyber security and enable us to innovate in ways we can't today.

For example:

- ✓ We would be able to measure the amount of security protection a given practice or product provides. We could see how that amount changes with adjustments we might make in how we perform that practice or configure that product. Then we'd be able to optimise our use of security solutions within the cost, manpower and time constraints we have to live under.
- ✓ We would be able to measure each of the threats we face, not just in terms of its level of activity (how much malware we are seeing, how many intrusion attacks, etc.) but in terms of its virulence, i.e. its capacity to cause harm. Just as, for Covid-19 (the Wuhan novel corona virus), the WHO knows that it is the virus' infectivity and mortality that it needs to measure and assess so it can provide guidance on protection measures, we would be able to work out, for each type of security threat we face, what the key parameters are that we need to measure so we can understand its capacity to cause harm and make informed decisions about how to protect against it.
- ✓ We could tailor our use of Best Practices or international standards according to our particular circumstances. We could ditch the 'one-size-fits-all' approach and adapt our chosen set of security practices to fit our technologies, our operational models, and our specific and sectoral threat environments. We need the ability to do this. A set of security practices that might work well for a small regional bank isn't likely to be sufficient for a major global bank. What works for a white goods manufacturer might not be sufficient for an aircraft manufacturer, and that is certainly not going to be what is needed by a national airline providing online check-in.

- ✓ We could adjust key security measures according to threat virulence levels as they change. Even outside the appearance of novel new attacks, a threat's virulence can change over time. By continuously monitoring threat virulence levels and understanding the amount of protection our key security measures provide, we could adjust those measures up or down as threat virulence levels change to maintain a required level of protection.
- ✓ Each organisation will have the ability to manage its cyber security risks with more confidence. CISOs will be able to provide Risk Committees (and Regulators) with objective measures of the risks the business is carrying. We will never get to the stage where there is just a single Risk number to be monitored and a single tap we can turn up or down to keep that number within a mandated range, but CISOs will be able to provide accountable Executives with a set of objectively measured metrics and all the information they need to make fully informed risk management decisions.

We currently practise cyber security as a craft. This makes outcomes uncertain and limits our ability to adapt and innovate. Instead, we should try treating cyber security as a science. Then we could measure the inputs (threats, vulnerabilities), measure the outputs (protection or risk), set and adjust our controls to get the outputs we desire given the inputs and resource constraints we are dealing with, and manage and control the whole process with transparency and confidence.

What would 'treating cyber security as a science' look like?

We do know at least one way to treat cyber security as a science. It is called TBSE, Threat-Based Security Engineering. It is possible that in years to come people will propose other ways to treat security as a science, but at the present time it is the only way (in the public domain) that we know for doing this.

TBSE is a paradigm, a conceptual way of thinking about what is going on between threats, vulnerabilities and controls when threats engage with a system and give rise to risk. This paradigm works on the basis of these interactions being stochastic rather than deterministic.

This point about the interactions being stochastic is key. Over the past five decades, many bright people have attempted to analyse security interactions deterministically. The reason why, despite all this effort, they have not been able to find any objective way to measure security protection or calculate security risk is that the processes that take place when risk is created are, at their core, stochastic. Deterministic analysis is simply the wrong way to go about analysing these dynamics to calculate results. It is a bit like trying to explain disease in terms of the imbalance of humours rather than infection by pathogens. It is fundamentally the wrong way to try to understand how the thing works.

I am not the first person to make this point. Indeed, there has been quite a bit of academic work done over the past twenty years, mostly under the heading "The Economics of Information Security" that adopts this perspective. Dozens of papers have been published showing how to analyse specific interactions non-deterministically to answer specific security questions. And at least one paper has shown how the three related analyses they discuss could be joined together like the carriages of a train to answer a slightly broader security question than those addressed by each of the three constituent parts.

However, none of these papers has provided a fully general framework for doing a broad spectrum of these types of analysis or shown how multiple links can be brought together into a chain that,

conceptually, can cover the whole process from the origin of a threat through to the operational outcomes and harms that that threat might cause.

This is what TBSE does, and is how it serves to turn cyber security into a science. TBSE is a stochastic paradigm that covers the full gamut of what goes on from the origin of a threat through to the material harms that threat causes. It provides a framework that allows us to analyse any risk-relevant interaction using stochastic modelling methods, and to calculate the effect any interaction has on the progress of a threat as it works its way towards causing harm. Being a paradigm rather than just an approach, TBSE shows us not just how to analyse individual risk-relevant interactions but how to combine a series of such analyses into a chain that, if followed to the end, provides an objective analytical result for the amount of risk a threat creates.

How would I use this?

Making use of TBSE does not have to be a huge task. It doesn't mean you have to stop doing any of the things you are already doing or replace practices and solutions you have invested in in the past. You can apply it one step at a time. You can apply it in a lite manner or in a more weighty analytical manner. And you do not need to take it any further or faster than you want.

Here are three suggestions for how you could try out TBSE to see what it can do for you.

Compliance assessments

One simple and quickly beneficial way of using TBSE is to start with just the paradigm itself, putting all the numerical aspects to one side.

Many organisations already operate security compliance schemes of one flavour or another. These schemes usually involve assessing either a system or an operational unit against a comprehensive catalogue of security measures (ISO27000; ISF; CIS; PCI; CSF; there are many to choose from) and coming up with a compliance score as a result. A directive somewhere will say what the threshold compliance score is for acceptability, and systems or units that score less than that threshold are usually required to implement a security improvement plan to get their score up over the line.

All these schemes have their problems.

- ✓ They are a burden on development teams. No matter how efficient you make your scheme, it will always be an overhead on their BAU operations. When delivery timescales and resources are tight, you will always get some sort of pushback.
- ✓ Their 'one size fits all' nature makes them less than a good fit. For any individual system you assess, there will always be some controls in the catalogue that don't apply. You have to work your way through the catalogue manually dropping each 'Not Applicable' control from the compliance assessment.
- ✓ They can generate discord. Especially if a system doesn't score well, you can get into arguments about what the assessment results mean. "Who came up with that number as the mandated

threshold result?" "Why is a score of 'threshold+1%' deemed acceptable when one of 'theshold-1%' isn't?"

- ✓ There is always the nagging question as to what does a compliance score tells you about security risk.

Part of the problem is the lack of any linkage connecting technical results to business goals. The technical result is a measurement of security implementation. The business goal is to reduce security risk to an acceptable level. How does the former tell you anything about the latter? The TBSE paradigm provides that linkage.

What this means in practice is you map each security measure in your catalogue according to how it contributes to reducing risk, i.e. where it fits in the paradigm. This is actually very straightforward, and it can bring out some interesting insights you might not have realised.

You work through your catalogue and map each security measure in several ways: against the threats it protects against; against the breaches it protects against; whether it is an 'upstream' control or a 'downstream' control (which is something you can do either coarsely or granularly); whether it sits in the middle of the stream or the edges (ditto); and so forth.

The assessment process can use this to tailor the scope of the assessment automatically to the needs of the system being assessed, and the assessment results will tell you not just an overall compliance number but from several different perspectives the extent and manner in which the controls implemented provide risk reduction.

You can do all of this using just the paradigm on its own. Because you are not using the numerical aspects of TBSE, it can't provide you with absolute numbers for residual risk. But it will provide you with a richer insight into the balance and coverage of the controls applied and, crucially, where the protection holes lie. For example: Is the system adequately protected against each class of security breach? Do you have good Defence in Depth or are there places where protection is only one not-well-implemented control deep? Given what controls you do have in place, what's the most efficient way to get the assessment results above the line?

If you also map your organisation's stated risk appetites / risk tolerances onto the TBSE paradigm, then you can show assessment results against those risk tolerances and see at a glance each way that a system is or isn't within tolerance. (This feature helps deal with much of the usual discord.)

As this is the simplest way to make use of the ideas that underpin TBSE, it is the most common type of TBSE engagement I undertake at present.

Threat monitoring

The next most straightforward way to use TBSE is to augment your controls monitoring with in-parallel threat monitoring.

Assessment schemes like those discussed in the preceding section are built on point-in-time assessments of control implementations. With threats becoming ever more sophisticated, we are seeing a move towards continuous

monitoring of critical controls. This provides early warnings when control implementations slip so corrective action can be taken promptly.

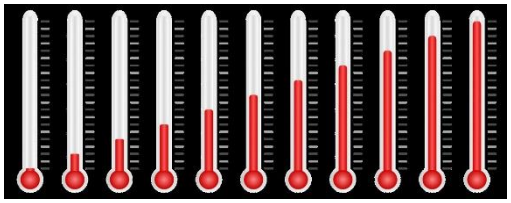
However, you still have the problem that you are measuring a control's implementation, not the amount of security protection it's providing. One of the first things TBSE can show you as you get to know it is that the way to get around that obstacle is to monitor the threat (or threats) a control is protecting you against in parallel with monitoring the control.

TBSE helps you see into the dynamics in play when a threat engages with a control. That tells you two things. It tells you how to measure the virulence of the threat, and it tells you how to measure the protection the control is providing. It shows you which aspects of the threat you need to measure (over and above simply its rate of attack) in order to understand its ability to cause you harm. And that then tells you straight away which aspects of the control you need to measure, over and above simply the extent to which it has been rolled out, to understand the level of protection it provides.

Getting simultaneous reports on both the virulence profile of the threat and the protection profile of the control lets you see what is really happening on your risk time line. You can distinguish between times when a threat's volume goes up but its virulence stays much the same (no need to take corrective action) and when its volume stays static but its virulence is going up (prompt action required).

Not only will your stakeholders really appreciate you being able to show them something meaningful about the organisation's risk but you will be able to show them what type of adjustments or response would be needed to stop the risk growing out of tolerance. That might be simply to put some fresh energy into completing the roll out of that control or to point out that the time has come for them to make an investment that they have been hesitant about supporting in the past.

Security metrics

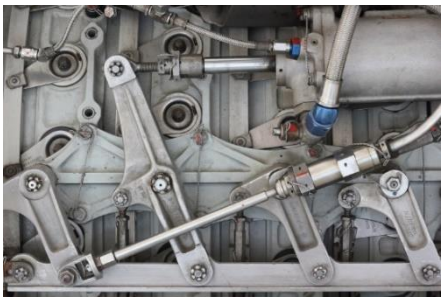


This next way to use TBSE entails using the paradigm a little more extensively. It is aimed at building a set of metrics that actually tell you the things you need to know to stay secure.

Many organisations already operate a number of security metrics. They measure various aspects of the implementation of their security solutions or activities, and report those measurements against policy to drive implementation improvements. This is known as "security verification". Verification is about showing that security controls are being applied as they should. It is not the same as "security validation" which is about showing that those controls are actually providing the desired amount of security protection.

Security verification serves as a proxy for security validation. It works on the premise that if security controls are being applied, operated and managed to the standard required by policy then one can presume they are providing security protection to the level desired by the owners of those policies. This is normally about as much as one can do. It leads to metrics that lend themselves to dashboard RAG diagrams and regular periodic tracking but it doesn't really tell you much about the things you need to know to stay secure.

To move beyond security verification to security validation requires a conceptual understanding of how security controls provide security protection. This is where TBSE comes in. You can take each individual component of your security armoury, apply the TBSE paradigm to create a conceptual understanding of the dynamics going on there, and from that you can identify what you would need to measure or analyse to answer the risk management questions you want your metrics to answer.



For each security component you want to explore, you use TBSE to build a relatively simple stochastic model for how that activity or solution provides security protection. That will show you what data you need to gather upstream of that component, what data downstream, and what data you need about the implementation or operation of the component itself. That will give you a small number, usually two three or four, of metrics to set up.

You will operate those metrics and gather that data to whatever level of precision you can achieve readily, and that will give you an initial understanding of the dynamics at work for that component.

That might be sufficient to show you what you need to change about the component's security design to make a worthwhile improvement in the amount of security it provides. And if not, then it will show you where you need to build in more detail (to the model or the data you gather) so you can answer the questions at hand. As with any type of modelling, the general rule of thumb is the more detail you have in the model and input data, the more detail you get back (and the more confidence you can have) in the results.

You can apply TBSE one component at a time. You take the analysis for that component as far as you need it to go to give the answers you want, you operationalise the measurement and reporting processes for the relevant metrics, and on you go. When ready, you move on to the next component in your security armoury. At your own speed, you evolve a new set of metrics that lend themselves to dashboard RAG diagrams and regular periodic tracking, except that this time they do help you understand the things you need to know to stay secure.

I have helped clients develop security metrics in this way across a range of security areas including: Resistance to malware; Resistance to Phishing; Staff security awareness; End-user password strength; Perimeter resistance to intrusion attacks; Software vulnerability removal; DDoS protection; and others.

All sorts of exciting opportunities await those willing to treat cyber security as a science. If you would like to talk about any of this in more detail, please get in touch. Email me at john.leach@jlis.co.uk or call 07734 311567 (+44 7734 311567).