UNIVERSITY OF LANCASTER

&

MITIGATE CYBER

KNOWLEDGE TRANSFER PARTNERSHIP № 11598

Threat Intelligence Analytical Software

Ben Goldsworthy

b.r.goldsworthy@lancaster.ac.uk

supervised by

Daniel Moore, Mitigate Cyber

Dr Daniel Prince, University of Lancaster

March 5, 2021

Version Control			
Version	**Date issued**	**Author**	**Comment**
v1.0	2021-03-05	Ben Goldsworthy	Initial release

Contents

1 Introduction

KTP № 11598 was a public-private research partnership between Lancaster University and Mitigate Cyber that ran from August 2019–March 2021, part-funded by Innovate UK. The goal of the partnership was to design and implement a novel IT threat intelligence analysis and quantitative risk calculation tool that would integrate into Mitigate Cyber's existing SaaSS platform. Unfortunately, the project was cancelled before it could achieve its goals of completing a prototype.

This document compiles all of the work completed as part of the project that is suitable for release into the public domain. Firstly, we detail the findings of a multivocal literature review into the current state-of-the-art in both threat intelligence and software development. Secondly, we present the design and implementation work that was completed prior to the project's cancellation, including the underlying theory and mathematics. Finally, we list the development roadmap we had intended to follow prior to the cancellation.

2 Literature Review

This section details some of the results of the first stage of the project: "Investigation & Understanding".

We first examine the current state-of-the-art in the field of threat intelligence. We achieve this via a multivocal literature review (MLR) covering the field of information security threat intelligence (ISTI) research,[1] existing products, relevant insights from the broader security and intelligence studies literature (including military and law enforcement sources) and a range of models proposed for each element of the threat intelligence process. From this, we then propose a framework for threat intelligence and compare these models against it.

Finally, from this we derive a reference architecture for such systems. In line with the terminology from the ISO/IEC 27000-series of standards, we define this system as a threat intelligence management system (TIMS).

2.1 Legislation and Standards

In this section, we detail the legislative environment in which a TIMS must operate. For brevity, we consider only legislation applicable within the UK. We then discuss relevant technical standards and how they may impact on the design of the solution.

2.1.1 Legislation

This subsection describes pieces of legislation that will have to be considered when drafting the requirements for any TIMS. This includes data protection legislation and information security regulations.

[1] We have tried to avoid using terms such as 'cyber threat intelligence' throughout the report, in favour of more precise terminology such as 'ISTI'. See Andrew Futter, '"Cyber" semantics: why we should retire the latest buzzword in security studies' (2018) 3(2) Journal of Cyber Policy 201, for further discussion of the issues with the term 'cyber'.

GDPR & DPA 2018

The EU's General Data Protection Regulation (GDPR) was signed into law on 2016-04-14 and came into effect on 2018-05-25. Unlike the previous EU Data Protection Directive,[2] the GDPR was directly enforceable and did not require implementation by Member States in the form of domestic legislation. The Regulation 'lays down rules relating to the protection of natural persons with regard to the processing of personal data and rules relating to the free movement of personal data.'[3]

The GDPR defines 'personal data' as 'any information relating to an identified or identifiable natural person ("data subject")', 'processing' as 'any operation or set of operations which is performed on personal data or on sets of personal data, whether or not by automated means, such as collection, recording, organisation, structuring, storage,...erasure or destruction' and a 'data controller' as the party which 'determines the purposes and means of the processing of personal data'.[4] It also defines 'special category' personal data, but this is not relevant to our discussion.

A TIMS would obviously not be much use without customers, so we can assume that any vendor will have to process and store personally-identifying customer data at the very least. This will make them the 'data controller'.

The GDPR lays out the principles relating to the processing of personal data:

(a) lawfulness, fairness and transparency;

(b) purpose limitation;

(c) data minimisation;

(d) accuracy;

(e) storage limitation;

(f) integrity and confidentiality; and

(g) accountability.[5]

Lawfulness, fairness and transparency

Lawful processing is defined in the GDPR as data processing to which 'at least one of the following applies':

(a) the data subject has given consent to the processing of his or her personal data for one or more specific purposes;

(b) processing is necessary for the performance of a contract to which the data subject is party or in order to take steps at the request of the data subject prior to entering into a contract;

[2] Directive on the protection of individuals with regard to the processing of personal data and on the free movement of such data 1995-11-23.

[3] Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) 2016-05-04, art. 1(1).

[4] ibid art. 4, paras. 1–2 and 7.

[5] ibid art. 5, paras. 1 and 2.

(c) processing is necessary for compliance with a legal obligation to which the controller is subject;

(d) processing is necessary in order to protect the vital interests of the data subject or of another natural person;

(e) processing is necessary for the performance of a task carried out in the public interest or in the exercise of official authority vested in the controller;

(f) processing is necessary for the purposes of the legitimate interests pursued by the controller or by a third party, except where such interests are overridden by the interests or fundamental rights and freedoms of the data subject which require protection of personal data, in particular where the data subject is a child.[6]

The GDPR also lays out the data subject's right to access their personal data and the responsibilities of the data controller to make information such as their identity and contact number, the contact details of the data protection office and the legal basis for any proposed processing clear in advance.[7]

Purpose limitation

With the exception of archival in the public interest, scientific or historial research or statistical purposes,[8] personal data must only be 'collected for specified, explicit and legitimate purposes and not further processed in a manner that is incompatible with those purposes'.

Data minimisation

Personal data must be 'adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed'.

Accuracy

Personal data must be 'accurate and, where necessary, kept up to date; every reasonable step must be taken to ensure that personal data that are inaccurate, having regard to the purposes for which they are processed, are erased or rectified without delay'. The data subject is considered to have rights to both rectification and erasure of their personal data.[9]

Storage limitation

Again notwithstanding the archival exemption, personal data must be 'kept in a form which permits identification of data subjects for no longer than is necessary for the purposes for which the personal data are processed'. Transfer to countries outside of the UK are limited to those that have been approved by the Secretary of State.[10] Re-identification of anonymised data is an offence.[11]

Integrity and confidentiality

Data must be kept secure whether during processing or at rest. This includes 'protection against unauthorised or unlawful processing and against accidental loss, destruction or damage, using appropriate technical or organisational measures.'

[6] GDPR, art. 6.
[7] ibid arts. 12–15.
[8] Data Protection Act 2018 2018, s. 19.
[9] GDPR, arts. 16–20.
[10] DPA 2018, s. 18.
[11] ibid s. 171.

The GDPR states that, '[i]n the case of a personal data breach, the controller shall without undue delay and, where feasible, not later than 72 hours after having become aware of it, notify the personal data breach to the supervisory authority...unless the personal data breach is unlikely to result in a risk to the rights and freedoms of natural persons.'[12] Infringements of the GDPR's provisions, depending on the specific provisions infringed, are 'subject to administrative fines up to 20,000,000 EUR [or] up to 4 % of the total worldwide annual turnover of the preceding financial year, whichever is higher'.[13]

The Data Protection Act 2018 (DPA 2018) serves to ensure that the provisions of the EU's GDPR are included in UK law in the event of that country leaving the EU, 'and applies a broadly equivalent regime to certain types of processing to which the GDPR does not apply.'[14] The Act covers 'the processing of personal data', and is largely a complete transplantation of the GDPR into UK law with additional provisions for matters outside of the scope of EU law (e.g., national security). However, there are some more subtle differences. For example, the GDPR states that the age at which a child can consent to data processing is sixteen, but that 'Member States may provide by law for a lower age...provided that such lower age is not below 13 years'[15]—the DPA 2018 opts for this lower limit.[16]

Generally, though, differences are minimal; the purpose of the Act is to ensure that the UK's data protection laws will be considered 'adequate' after leaving the EU, allowing the transfer of EU citizens' data to and from the UK.[17] Note, also, that the UK's exit from the EU will not affect UK businesses' need to comply with the regulations when operating internationally, as they apply to anybody who processes the data of EU citizens, regardless of where such processing occurs geographically.

NIS Directive & NIS Regulations

The Network Information Security Directive (NIS Directive) entered force in mid-2016.[18] As a directive, rather than a regulation, it did not create an enforceable law, but rather directed all EU Member States to implement it in the form of national laws by mid-2018. The UK duly issued the Network and Information Systems Regulations 2018 (NIS Regulations) statutory instrument, which came into force on 2018-06-20.[19]

The stated purpose of both the Directive and the Regulations is to '[lay] down measures with a view to achieving a high common level of security of network and information systems within the Union so as to improve the functioning of the internal market.'[20] As a result, the government was required to designate a number of 'national competent authorities',[21] a 'single point of contact on the security of network and information systems'[22] and a computer security incident response team (CSIRT). The national competent authorities vary by sector, and the National Cyber Security Centre (NCSC) was appointed as the single point of contact and CSIRT for the UK.

[12] GDPR, art. 33(1).
[13] ibid art. 83(4–5).
[14] DPA 2018, s. 1(3).
[15] GDPR, art. 8(1).
[16] DPA 2018, s. 9(a).
[17] GDPR, art. 45.
[18] Directive (EU) 2016/1148 of the European Parliament and of the Council of 6 July 2016 concerning measures for a high common level of security of network and information systems across the Union 2016-07-19.
[19] Network and Information Systems Regulations 2018, SI 2004/3166 2018.
[20] NIS Directive, art. 1, para. 1.
[21] NIS Regulations, reg. 3.
[22] ibid reg. 4, r. 1.

Part 2 of the Regulations serves to identify who the UK government considers to be operators of essential services (OESs) and digital service providers (DSPs).

The term OES covers:

> (a) an entity provid[ing] a service which is essential for the maintenance of critical societal and/or economic activities;
>
> (b) [where] the provision of that service depends on network and information systems; and
>
> (c) [where] and incident would have significant disruptive effects on the provision of that service.[23]

A DSP is 'any person who provides: (a) online marketplace; (b) online search engine; (c) cloud computing service.'[24] They are beholden to the Information Commissioner's Office (ICO).

Designated competent authorities are responsible for establishing the threshold for essential services within their sector. Anyone crossing this threshold 'must notify the designated competent authority of that fact'.[25] A designated authority can also designate organisations who do not cross the set threshold as OES' at their discretion.[26] In short, the NIS Directive applies to organisations that are otherwise considered to be part of the country's critical national infrastructure (CNI).

Both OES' and DSPs have a duty to 'take appropriate and proportionate technical and organisational measures to manage risks posed to the security of the network and information systems on which their essential service relies' and to 'take appropriate and proportionate measures to prevent and minimise the impact of incidents affecting the security of the network and information systems used for the provision of an essential service, with a view to ensuring the continuity of those services.'[27] There is also an echo of the GDPR in the form of a 72-hour window in which the designated competent authority/ICO must be notified of 'any incident which has a significant impact on the continuity of the essential service which that OES provides'.[28]

In the event that the designated competent authority/ICO believes that an OES or DSP has failed to fulfil their security duties or comply with their notification requirements, they may issue an enforcement notice against that organisation. Fines to be levied start at a maximum of £1,000,000 'for any contravention which the enforcement authority determines could not cause a [network and information systems] incident' up to a maximum of £17,000,000 'for a material contravention which the enforcement authority determines has caused, or could cause, an incident resulting in an immediate threat to life or significant adverse impact on the United Kingdom economy.'[29]

Note that, whilst in the Regulations the obligations and punishments are the same for both OES' and DSPs, the Directive does state that '[i]n practice, the degree of risk for operators of

[23] NIS Regulations, art. 5, para. 2.
[24] ibid reg. 1, r. 2.
[25] ibid reg. 8, r. 2.
[26] ibid reg. 8, rr. 3–7.
[27] ibid reg. 10, rr. 1–2 & reg. 12, rr. 1–2.
[28] ibid reg. 11 & reg. 12, r. 3–6.
[29] ibid regs. 17–19.

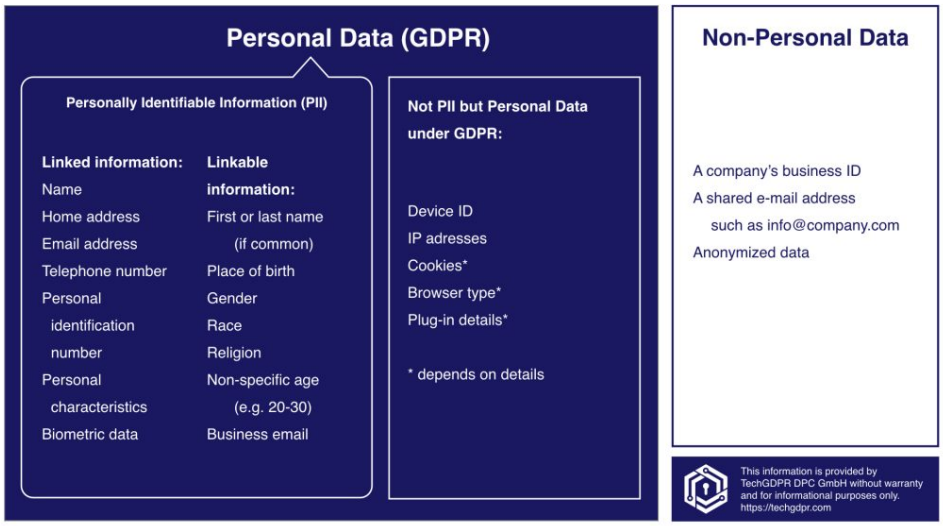


Figure 1: PII and personal data comparison.[31]

essential services, which are often essential for the maintenance of critical societal and economic activities, is higher than for digital service providers [and that, t]herefore, the security requirements for digital service providers should be lighter.'[30]

As the CNI sector (particularly healthcare) is a key target vertical for most information security organisations, the NIS Directive and NIS Regulations will need to be taken into account in the design of any product for this sector.

2.1.2 Standards

This subsection examines a pair of ISO/IEC standards: ISO/IEC 29100, which outlines a privacy framework very similar to the GDPR; and the ISO/IEC 27000-series, which describes how organisations can create and maintain an information security management system (ISMS). A TIMS would be a good fit within a larger ISMS, so this examination can provide useful information about the requirements that will be imposed on the system.

ISO/IEC 29100

This standard outlines a privacy framework that echoes many of the principles outlines in the GDPR above, although there are some key differences. For example, the term personally-identifiable information (PII) is used throughout in place of personal data, which covers a less expansive range of data; per the US Government's Office of Privacy and Open Government:

> [t]he term personally-identifiable information refers to information which can be used to distinguish or trace an individual's identity, such as their name, social security number, biometric records, etc. alone, or when combined with other personal or identifying information which is linked or linkable to a specific individual, such as date and place of birth, mother's maiden name, etc.[32]

To see how this compares with what is covered under the GDPR's definition of 'personal

[30] NIS Directive, recital 49.

[32] Office of Privacy and Open Government, The, 'Safeguarding Information' ⟨http://www.osec.doc.gov/opog/privacy/PII_BII.html⟩ accessed 3 December 2019.

ISO/IEC 29100 principle	GDPR principle
Consent and choice	Lawfulness, fairness and transparency
Purpose legitimacy and specification	Purpose limitation
Collection limitation	Purpose limitation
Data minimisation	Data minimisation
Use, retention and disclosure limitation	Storage limitation
Accuracy and quality	Accuracy
Openness, transparency and notice	Lawfulness, fairness and transparency
Individual participation and access	Lawfulness, fairness and transparency
Accountability	Accountability
Information security	Integrity and confidentiality
Privacy compliance	Lawfulness, fairness and transparency

Figure 2: Alignment of ISO/IEC 29100 and GDPR privacy principles.

ISO/IEC 29100 concepts	Correspondence with ISO/IEC 27000 concepts
Privacy stakeholder	Stakeholder
PII	Information asset
Privacy breach	Information security incident
Privacy control	Control
Privacy risk	Risk
Privacy risk management	Risk management
Privacy safeguarding requirements	Control objectives

Figure 3: Matching ISO/IEC 29100 concepts to ISO/IEC 27000-series concepts.[35]

data', see fig. 1.

For an example of an area in which the standard hews more closely to the GDPR, take the roles set out in § 4.2:

- PII principles, analogous to the GDPR's data subjects;
- PII controllers, analogous to the GDPR's data controllers;
- PII processors, analogous to the GDPR's data processors; and
- Third parties, who will generally 'become...PII controller[s] in [their] own right[s] once [they have] received the PII in question.'[33]

The standard also presents eleven privacy principles,[34] which largely overlap with those of the GDPR (see fig. 2).

[33] ISO/IEC 29100:2011: Information technology — Security techniques — Privacy framework (2011) pp. 5–6.
[34] ibid pp. 14–19.

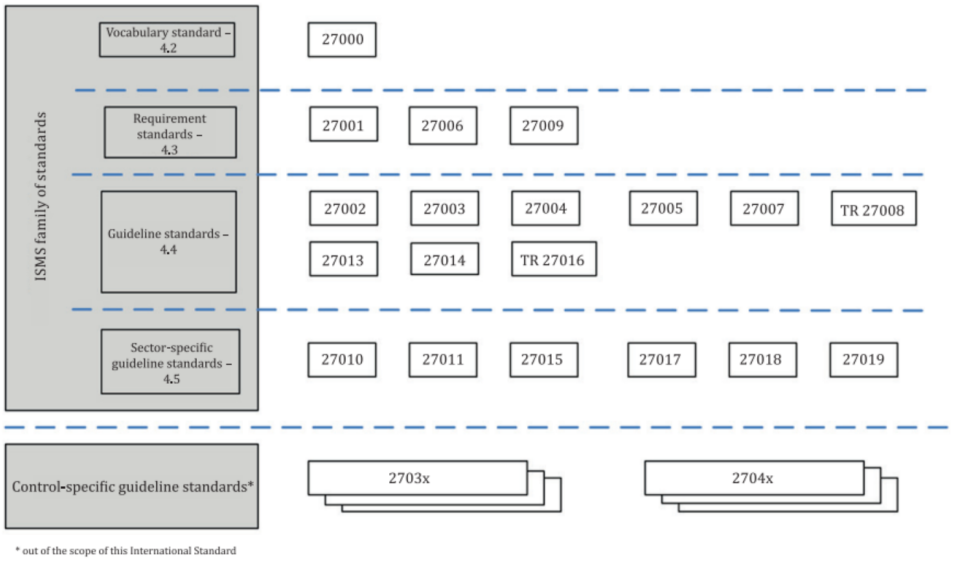


Figure 4: ISO/IEC 27000 family of standards.[36]

As our need to comply with the more demanding GDPR means that we will, as a by-product also satisfy ISO/IEC 29100, we shall not discuss it further, except to highlight the table in Annex A of the standard (see fig. 3) which matches ISO/IEC 29100 concepts with those from the ISO/IEC 27000-series (see fig. 3). As many of these ISO/IEC 29100 concepts have analogues within the GDPR, this can still be of use to us.

ISO/IEC 27000-series

The ISO/IEC 27000-series of standards, jointly issued by the ISO and IEC, describe the requirements of an ISMS, which is defined as follows:

> An information security management system (ISMS) consists of the policies, procedures, guidelines, and associated resources and activities, collectively managed by an organization, in the pursuit of protecting its information assets. An ISMS is a systematic approach for establishing, implementing, operating, monitoring, reviewing, maintaining and improving an organization's information security to achieve business objectives. It is based upon a risk assessment and the organization's risk acceptance levels designed to effectively treat and manage risks.[37]

There are currently 46 standards in the series, covering topics ranging from risk management (ISO/IEC 27005) to information security for the healthcare sector (ISO 27799) and divided into four groups:

- standards describing an overview and terminology (comprising solely ISO/IEC 27000);
- standards specifying requirements;
- standards describing general guidelines; and
- standards describing sector-specific guidelines.

[37] ISO/IEC 27001:2013: Information technology — Security techniques — Information security management systems — Requirements (2013) p. 14.

Fig. 4 illustrates these delineations.

Not all of these standards are relevant to our project. As such, we shall limit our investigation to the ISO/IEC 27001 and ISO/IEC 27002 documents, along with several guideline documents. We shall also only be referring to the most current versions of these standards (e.g., 27001:2013).

ISO/IEC 27001

ISO/IEC 27001's purpose, as laid out in its introduction, is as follows:

> to provide requirements for establishing, implementing, maintaining and continually improving an information security management system [which] preserves the confidentiality, integrity and availability of information by applying a risk management process and gives confidence to interested parties that risks are adequately managed.[38]

The document is divided into ten clauses, supplemented by a long annex of controls. The clauses provide an organisation with guidance on how they could construct and manage their ISMS, and are as follows:

1. Scope;
2. Normative references;
3. Terms and definitions;
4. Context of organization;
5. Leadership;
6. Planning;
7. Support;
8. Operation;
9. Performance evaluation; and
10. Improvement

The standard requires that any organisation's management 'demonstrate leadership and commitment with respect to the information security management system', through such imperatives as 'ensuring the information security policy and the information security objectives are established' and 'ensuring that the information security management system achieves its intended outcome(s)'.[39] There are also requirements provided for the creation and dissemination of information security policies, such as that 'information security policy shall be available to interested parties, as appropriate'.[40]

[38] ISO/IEC 27001:2013 (n 37) p. v.
[39] ibid p. 2.
[40] ibid p. 3.

The latest edition of the standard provides a list of 114 controls, grouped into 14 clauses and 35 categories, against which any proposed ISMS can be audited. Unlike the previous 27001:2005 edition, 27001:2013 drops both the emphasis on the Plan-Do-Check-Act (PDCA) cycle and the requirement that businesses use the controls listed in Annex A of the standard. Instead, alternate continuous improvement processes and control sets can be used, at the discretion of the organisation in question.

Guideline Standards

Amongst the various general and sector-specific guidelines, there are a number that may be of interest to our project.

- ISO/IEC 27002 'provides guidance on the implementation of information security controls';[41]
- ISO/IEC 27003 'provides a process-oriented approach to the successful implementation of the ISMS in accordance with ISO/IEC 27001';[42]
- ISO/IEC 27004 'provides guidance and advice on the development and use of measurements in order to assess the effectiveness of ISMS, control objectives, and controls used to implement and manage information security';[43]
- ISO/IEC 27005 'provides guidance on implementing a process-oriented risk management approach';[44]
- ISO/IEC 27010 'provides guidelines...for implementing information security management within information security communities...applicable to all forms of exchange and sharing of sensitive information, nationally and internationally, within the same industry or market sector or between sectors'[45]—as a crucial component of an ISTI is the ability to share information, as highlighted by the heavy focus on this area within the academic literature, we believe that this standard will be particularly relevant to our project and deserves further analysis;
- ISO/IEC 27017 'gives guidelines for information security controls applicable to the provision and use of cloud services';[46]
- ISO/IEC 27018 provides guidelines for organisations that 'provide information processing services as PII processors via cloud computing under contract to other organizations';[47]
- ISO/IEC 27799 'provides guidelines supporting the implementation of information security management in health organizations'[48]—as healthcare is a key target vertical for most information security organisations, we believe this standard to be relevant.

[41] ISO/IEC 27000:2016: Information technology — Security techniques — Information security management systems — Overview and vocabulary (2016) p. 23.
[42] ibid p. 23.
[43] ibid p. 23.
[44] ibid p. 23.
[45] ibid p. 25.
[46] ibid pp. 25–26.
[47] ibid pp. 26.
[48] ibid pp. 26–27.

ISO/IEC 27000-series and Our TIMS

We believe that a TIMS is a valuable 'associated resource', and the process of exploiting such intelligence is an 'associated activity', of a potential customer's ISMS. We base this view on the following sections of the ISO/IEC 27001 standard:

- 5.1(c) of the ISO/IEC 27001 standard requires that 'top management shall demonstrate leadership and commitment with respect to the ISMS by...ensuring that the resources needed for the ISMS are available';[49]
- 5.1(d) requires that they 'communicat[e] the importance of effective information security management';[50]
- 6.1.2 requires that an organisation 'define and apply an information security risk assessment process that...identifies the information security risks[,] analyses the information security risks [and] evaluates the information security risks', which clearly requires some form of threat intelligence to perform effectively;[51]
- 7.3(c) requires that 'people doing work under the organization's control shall be aware of...the implications of not conforming with the ISMS requirements';[52] and
- 7.4 requires organisations to 'determine the need for internal and external communications relevant to the ISMS'.[53]

In addition, should a vendor wish to include their software projects within the scope of an organisational ISO/IEC 27001 certification, any TIMS must both fit into a potential customer's ISMS and fulfil the vendor's own responsibilities.

As previously mentioned, ISO/IEC 27010 provides guidelines for applying ISO/IEC 27001 within information sharing communities. This can present difficulties, as 'part of the member organization will be within scope of the community ISMS and part will be outside' and 'members of the information sharing community may have their own information security management systems and, in consequence, some processes might fall within scope of both the community and members' management systems.'[54]

The standard describes sensitive communications in terms of three participants:

- the source of an item of information, who 'does not need to be a member of the community';
- the originator, who initiates the distribution within the community, who 'may, but need not be, the same as the source of the information' and who 'may conceal the identity of the source'; and
- the recipient(s) who receive the information distributed within the community, but who do not need to be members of the community.[55]

[49] ISO/IEC 27001:2013 (n 37) p. 2.
[50] ibid p. 2.
[51] ibid p. 3–4.
[52] ibid p. 5–6.
[53] ibid p. 6.
[54] ISO/IEC 27010:2015: Information technology — Security techniques — Information security management for inter-sector and inter-organizational communications (2015) p. 3.
[55] ibid p. 4.

The guidelines augment a few of the controls listed in ISO/IEC 27001, for example by highlighting that '[i]nformation provided by other members of an information sharing community is an asset' and should be treated as such and that information classification should also take into account credibility and priority.[56] It also introduces an additional control objective to clause 8 (asset management) aimed at providing information exchange protection, such as by ensuring source anonymity and displaying information disclaimers to recipients.[57]

Interestingly, the guidelines also state that '[i]nformation sharing communities should consider implementing alternative mechanisms for information sharing that do not rely on electronic messaging, and enabling members to specify that specific messages are distributed by such other routes.'[58]

§ 8.4.7 addresses the onward release of received information, citing the Traffic Light Protocol (TLP) as an option for 'indicat[ing] how information can be further distributed without seeking additional approval'.[59] The Protocol is described in Annex C of the standard.[60] 'The TLP is based on the concept of the originator labelling information with one of four colours to indicate what further dissemination, if any, can be undertaken by the recipient'—the colours used in the standard are:

- RED - Personal for Named Recipients Only;
- AMBER - Limited Distribution;
- GREEN - Community Wide; and
- WHITE - Unlimited (subject to standard copyright rules).[61]

The standard also states that '[a]ll sensitive information will be deemed to be AMBER unless otherwise stated or written', however 'the identity of the source of the sensitive information will always be RED.'[62]

ISO/IEC 27017 addresses the use of cloud services, on the basis that '[t]he provision and use of cloud services is a kind of supplier relationship, where the cloud service customer is an acquirer, and the cloud service provider is a supplier' and so clause 15 applies. It then provides implementation guidance for each control in Annex A, as well as a few additional ones, for both the cloud service customer and the cloud service provider.

Appendix A lists all of the ISO/IEC 27000-series controls that we believe to be relevant to this project.

2.2 Current State-of-the-Art

This section presents a multivocal literature review (MLR) of the contemporary state of the ISTI research field and market. First, we review the academic literature to present a top-level view of the state-of-the-art. Secondly we assess the current state of the ISTI market. Thirdly, we detail a range of potential data sources that an TIMS may choose to leverage.

[56] ISO/IEC 27010:2015 (n 54) p. 6.
[57] ibid p. 7–9.
[58] ibid p. 11.
[59] ibid p. 9.
[60] ibid p. 25.
[61] ibid p. 25.
[62] ibid p. 25.

2.2.1 Methodology

MLRs are a subset of systematic literature reviews (SLRs), which originated within the medical sciences, but which were introduced to the software engineering literature in the mid-2000s.[63] MLRs have begun to spread further within the information technology (IT) research community, including at least one paper on 'cyber' threat intelligence,[64] but adoption is currently limited.

The goal of an MLR is to conduct a literature review that encompasses both the academic literature and the 'grey literature (GL)' (alternatively 'practitioner' or 'non-academic literature') in a structured way that can account for the potential shortcomings of the latter. A set of guidelines exist for conducting MLRs in software engineering, which we find can apply to our own project without amendment.[65]

The methodology we have used for our MLR, adapted from,[66] is as follows:

1. Identify any existing reviews and plan/execute the MLR to explicitly provide usefulness for its intended audience.
2. Justify the decision to include the GL in the review.
3. Based on the research goal and target audience, define the research questions (RQs) (and be explicit about the type of each RQ).
4. Identify the relevant GL types and/or producers for the review.
5. Establish stopping criteria for your search.
6. Define the source inclusion/exclusion criteria.
7. Search for sources, up to the established stopping criteria.
8. Filter the sources by the established inclusion/exclusion criteria.
9. Perform quality assessment of the remaining sources, using a formal quality assessment checklist for GL sources. Establish a threshold for inclusion and filter out those sources that fall below this.
10. Extract data from the remaining sources.
11. Perform data synthesis on the extracted data.
12. Write a report on your review for the intended audience(s).

[63] Vahid Garousi, Michael Felderer, and Mika V Mäntylä, 'Guidelines for including grey literature and conducting multivocal literature reviews in software engineering' (2019) 106 Information and Software Technology 101, p. 1.

[64] Clemens Sauerwein and others, 'Threat intelligence sharing platforms: An exploratory study of software vendors and research perspectives' [2017] .

[65] Garousi, Felderer, and Mäntylä (n 63).

[66] ibid.

2.2.2 Literature Review

This subsection covers the current state-of-the-art within the academic and the grey ISTI literature. It begins with a brief look into the wider intelligence studies literature, followed by a meta-review of previously-published ISTI reviews and ending with coverage of various proposed models for various sub-areas of ISTI.

Within the world of IT, threat intelligence is a relatively new field. Elmellas describes it 'c[oming] to prominence in the mid-2000s as a solution to analysing and filtering data about emerging threats from several sources in real time to address the ever-increasing cyberthreat landscape.'[67] As shown in fig. 7, whilst there has been an explosion in activity within the field in recent years, this has not been reflected in the rate of academic publication. As such, there are few existing reviews of the literature and many of these reviews focus only on one sub-problem, such as threat intelligence sharing. For example, the Cyber Security Body of Knowledge (CyBoK) devotes only a single section to the topic, briefly discussing the use of honeypots and the definition of indicators of compromise (IoCs) whilst again focusing primarily on information sharing.[68]

Intelligence Studies

The field of intelligence studies (IS) is concerned with how the intelligence process operates in the abstract. Whilst it is not directly applicable to much of ISTI, there is still value in examining the findings of this field; not least of which is the fact that it represents many decades more research than does ISTI.

Gill and Phythian provide a comprehensive survey of the history of IS.[69] The authors define four main areas of IS work:

- research/historical;
- definitional/methodological;
- organizational/functional; and
- governance/policy.

They then trace the historical development of the field of IS in various countries, from the predominantly historical 'British school' that 'reflects not just the strength of the British community of historians but also that the two twentieth-century world wards provided much of the original raw material, as the strength of official secrecy ensured little on peacetime intelligence emerged before the 1990s and made the study of contemporary intelligence developments almost impossible' to the US IS community's greater focus on the definitional and organisational aspects—'more ink has probably been spilt on the Central Intelligence Agency (CIA) than on any other agency in the world', the authors write.[70]

The authors define 'intelligence' as:

[67] Jamal Elmellas, 'Knowledge is power: the evolution of threat intelligence' (2016) 2016(7) Computer Fraud & Security 5, p. 5.
[68] Hervé Debar, *Security Operations & Incident Management* (, CyBoK 2019) p. 28.
[69] Peter Gill and Mark Phythian, *Intelligence in an Insecure World* (Second, Polity Press 2012).
[70] ibid p. 4.

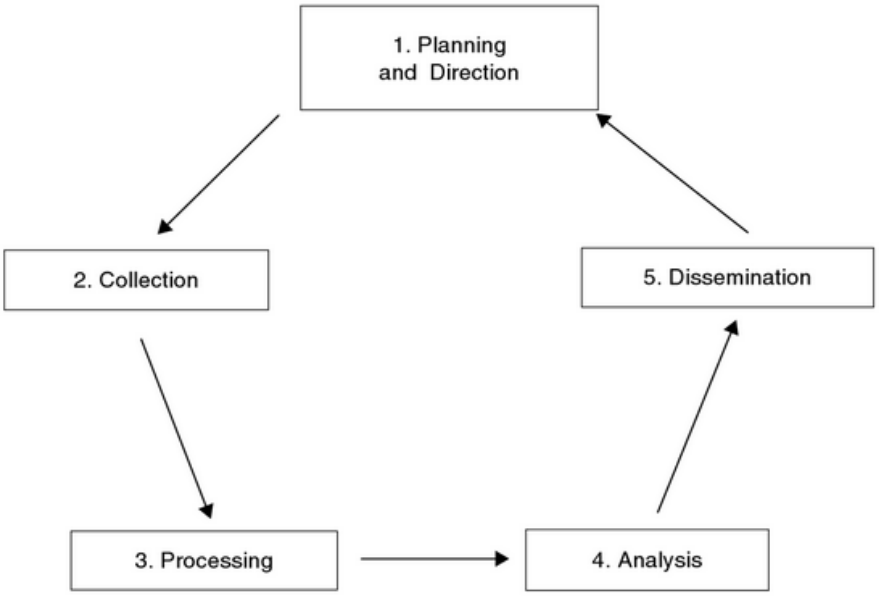


Figure 5: The intelligence cycle.[72]

> the mainly secret activities – targeting, collection, analysis, dissemination and action – intended to enhance security and/or maintain power relative to competitors by forewarning of threats and opportunities.[71]

Traditionally, the process of creating intelligence has been described as the 'intelligence cycle'.[73] This circular model (see fig. 5) comprises the following stages:

1. Planning and direction, which covers 'management of the entire effort, from identifying the need for data to delivering an intelligence product to a consumer';
2. Collection, or 'the gathering of the raw information needed to produce finished intelligence';
3. Processing, in which 'the vast amount of information collected [is converted into] a form usable by analysts through decryption, language translations, and data reduction';
4. Analysis, or 'the conversion of basic information into finished intelligence'; and
5. Dissemination, 'the distribution of the finished intelligence to the consumers' (and ' which logically feeds into the first').

As Gill and Phythian write, however, 'in recent years the utility of the concept of the intelligence cycle has been called into question'.[74] Hulnick suggests that 'the cyclical pattern does not describe what really happens', as '[p]olicy officials rarely give collection guidance[, c]ollection and analysis...in fact work more properly in parallel [and] the idea that decision

[71] Peter Gill, 'Theories of intelligence: Where are we, where should we go and how might we proceed?' in *Intelligence Theory: Key Questions and Deabtes* (Routledge 2008) p. 214.
[73] CIA, The Intelligence Cycle (2001).
[74] Gill and Phythian (n **??**) p. 12.

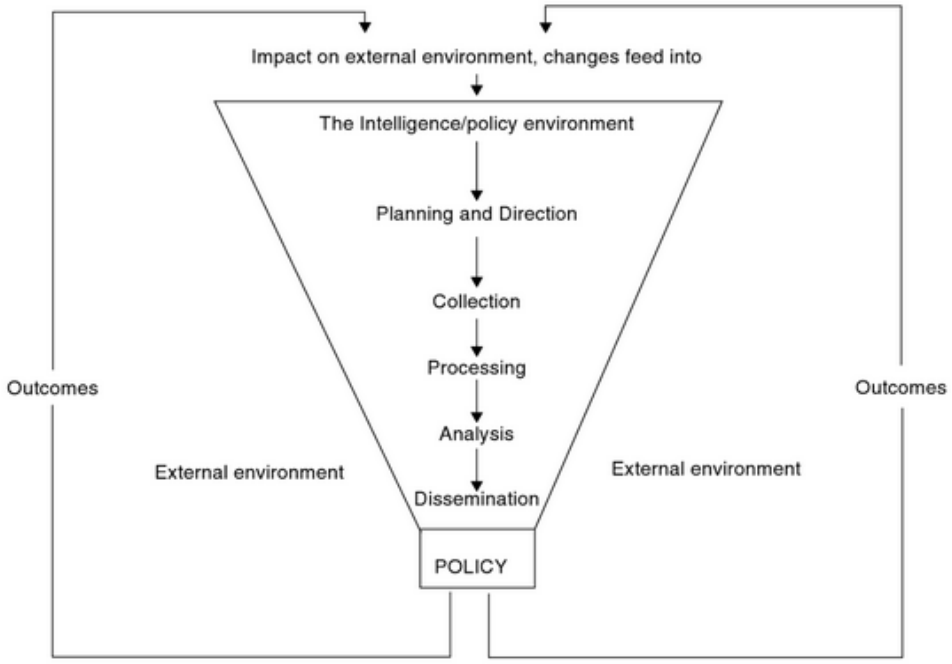


Figure 6: The intelligence process.[76]

makers wait for the delivery of intelligence before making policy decisions is equally incorrect.'[75] In addition, the intelligence cycle 'fails to consider either counter-intelligence or covert action.'

Gill and Phythian propose the model of the 'funnel of causality' (see fig. 6) as better capturing 'dynamic nature of intelligence's impact on the external environment'.[77] The model builds on earlier work by Wittkopf, Jones, and Kegley Jr.[78]

Others propose the Find, Fix, Finish, Exploit, Analyse and Disseminate (F3EAD) model more commonly used in military and counterterrorism operations.[79] This model extends the intelligence process to include actions resulting from the findings[80]—what Gill and Phythian refer to as 'power', contrasting it with 'knowledge'.[81]

The F3EAD model comprises the following steps:

1. Find, where a target is identified (using the five 'W's—'who, what, when, where, why');
2. Fix, where the previously-identified target is verified;
3. Finish, where will is imposed on the target;
4. Exploit, where evidence generated by the Finish phase is deconstructed;

[75] Arthur S Hulnick, 'What's wrong with the Intelligence Cycle' (2006) 21(6) Intelligence and national Security 959.

[77] Gill and Phythian (n **??**) pp. 13–14.

[78] Eugene R Wittkopf, Christopher M Jones, and Charles W Kegley Jr, *American Foreign Policy: Pattern and Process* (Cengage Learning 2007).

[79] Wilson Bautista Jr, *Practical cyber intelligence: how action-based intelligence can be an effective response to incidents* (Packt Publishing Ltd 2018) p. 122.

[80] Gill and Phythian (n **??**) p. 122.

[81] ibid p. 44.

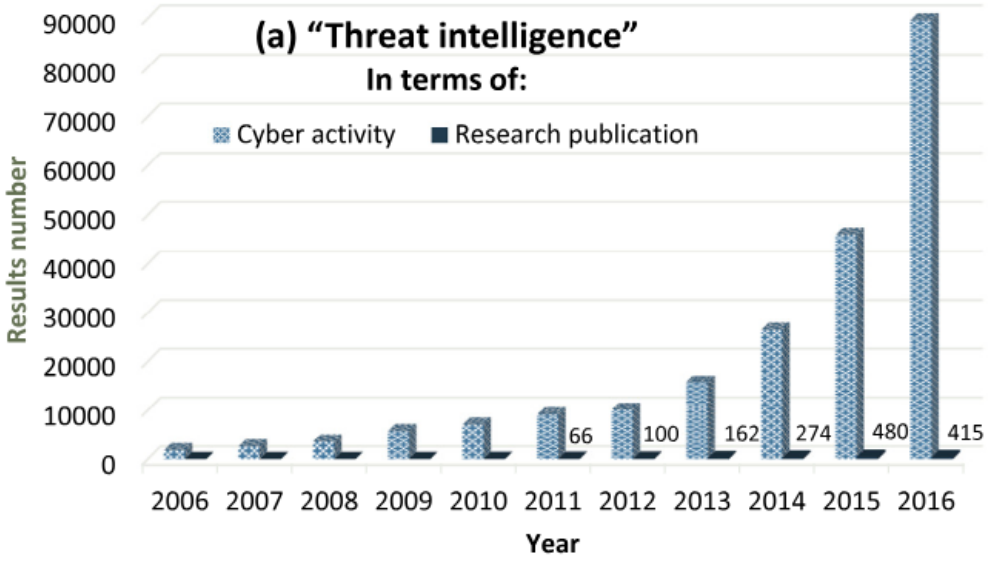


Figure 7: Trend of 'threat intelligence' in 'cyber activity', 2006–2016.[82]

5. Analyse, where the evidence thus generated is incorporated into the wider intelligence picture; and
6. Dissemination, where the results are published.

Whilst this model may have utility in the study and planning of offensive information security operations, such as the identification and breaking-up of organised malware gangs, we do not believe it is relevant to the vast majority of TIMS'. Only a very small minority of organisations, either state or commercial, will have the interest in thoroughly investigating attacks for attribution, and even fewer of these will be able to impose their will on the targets when identified; these are usually limited to state intelligence or policing agencies, but even they are often impotent in the face of trans-jurisdictional attacks.

As a result, we believe that the intelligence funnel concept provides the best way of thinking about general ISTI, and as such is the most useful model to apply to the design of our own TIMS.

Meta-review

The most comprehensive survey paper of the field of ISTI as a whole that we found is Tounsi and Rais'. The authors lament that 'few researches have been done to examine and identify characteristics of ISTI and its related issues', identifying that 'a significant body of work has been dedicated to threat intelligence sharing issues [whilst] in contrast, less research has been devoted to areas like technical threat intelligence (TTI) problems and how to mitigate them.'[84]

The authors divide ISTI into four sub-domains:

- strategic threat intelligence, high-level information consumed by decision-makers;
- operational threat intelligence, information about specific impending attacks consumed by higher-level security staff;

[84]Wiem Tounsi and Helmi Rais, 'A survey on technical threat intelligence in the age of sophisticated cyber attacks' (2018) 72 Computers & security 212, p. 216.

Table 2 – Reasons for not to share.

1	Fearing negative publicity
2	Legal rules, Privacy issues
3	Quality issues
4	Untrusted participants
5	Believing that the incident is not worth to share
6	Budgeting issues
7	Natural instinct to not to share
8	Changing nature of cyber attacks
9	Unawareness of the victimized organization about a cyber incident
10	Believing that there is a little chance of successful prosecution

Figure 8: Reasons not to share TI data.[83]

- tactical threat intelligence, information about how threat actors are conducting attacks consumed by incident responders; and
- technical threat intelligence, information normally consumed through technical resources.[85]

Strategic and tactical threat intelligence are identified for long-term use, whilst operation and technical threat intelligence are for short-term or immediate use. The authors also identify ten key reasons for organisations not to share ISTI data with one another; see fig. 8.

We found a number of reviews of specific areas of ISTI. Fachkha and Debbabi survey the use of darknet—'traffic targeting advertised, but unused, Internet Protocol (IP) addresses'—and honeypot monitoring in collecting ISTI, bolstered by case studies of various worms and botnets.[86]

Within the most heavily-researched area of ISTI sharing, Burger and others present a five-layer taxonomy for classifying exchange technologies (see fig. 9) whilst Sauerwein and others found that a standardised format for representing ISTI is currently lacking (whilst they do accept that Structured Threat Information eXpression (STIX™) appears to be the *de facto* standard, 'most platforms do not fully utilize its descriptive capabilities'), that most existing TIMS' are closed-source, that the research focus is on data collection rather than data analysis and that the degree of automation currently on offer is insufficient. Burger and others

Wagner and others provide a comprehensive survey of the state of ISTI sharing technologies currently existing. They again highlight the need for more automation, writing that 'current sharing methods are heavily based on manual input and therefore labor intensive.' They also identify three common sharing models: peer to peer; peer to repository; and hybrid, which combines the previous two models. They offer eleven characteristics of actionable ISTI (see fig. 10) as well as an overview of the regulations surrounding ISTI sharing, emphasising the need for international alignment as 'cyber attacks do not know any borders, therefore, CTI sharing should ideally not be impeded by various country regulations'. Interestingly, the

[85] Tounsi and Rais (n 84) p. 215.

[86] Claude Fachkha and Mourad Debbabi, 'Darknet as a source of cyber intelligence: Survey, taxonomy, and characterization' (2015) 18(2) IEEE Communications Surveys & Tutorials 1197.

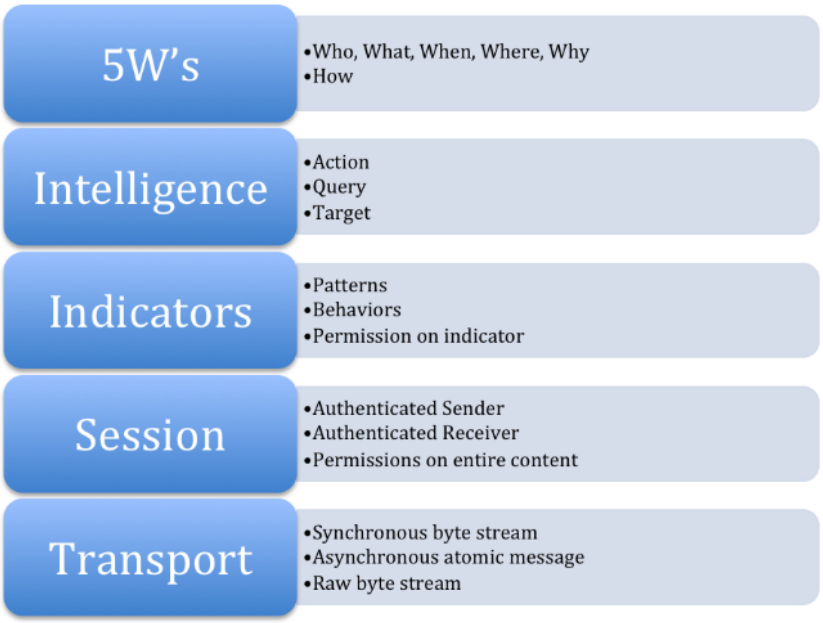


Figure 9: Five-layer taxonomy of ISTI exchange technologies.[87]

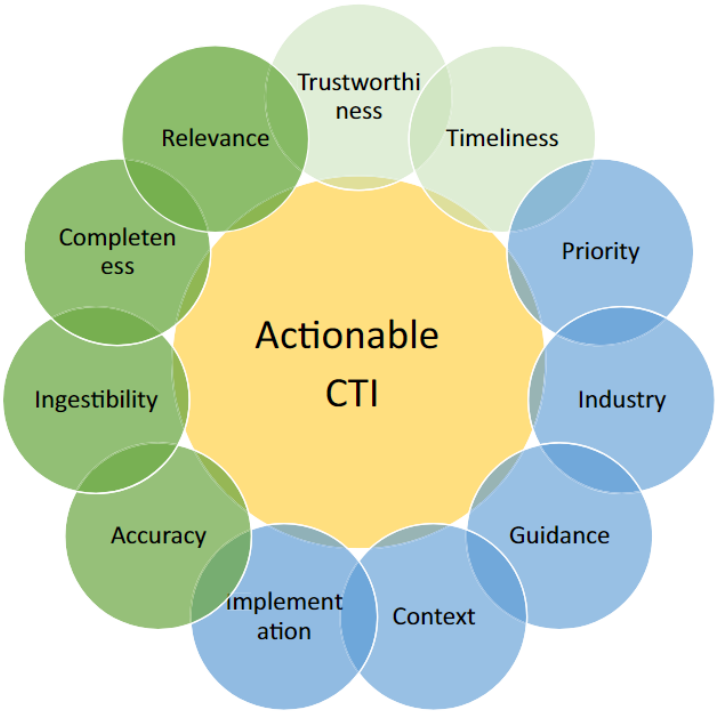


Figure 10: Characteristics of actionable ISTI.[88]

Figure 11: The primary dimensions of information sharing.[89]

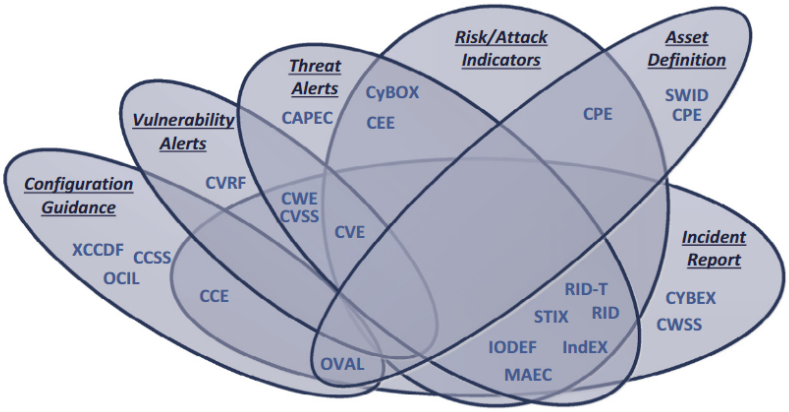


Figure 12: Knowledge areas covered by different existing ISTI sharing standards.[91]

authors warn that 'it may become obligatory for organizations to have a threat intelligence program...and share their information [with] stakeholders...held responsible in the future for not sharing known threats that affected others and results in a breach.' Unfortunately, this appears to be conjecture on the part of the authors and no evidence is provided in support of the claims.[90]

Skopik, Settanni, and Fiedler describe information sharing in terms of five primary dimensions (see fig. 11) and elaborate on each, addressing key legislation such as the EU's NIS Directive and the US' White House Executive Order (EO) 13636 and Presidential Policy Directive (PPD) 21[92] as well as standardisation efforts. They also highlight the knowledge areas covered by existing ISTI sharing standards (see fig. 12), demonstrating a lack of any fully-comprehensive alternative at present.[93]

[90] Thomas D Wagner and others, 'Cyber threat intelligence sharing: Survey and research directions' (2019) 87 Computers & Security 101589.

[92] NIS Directive; White House, The, Executive Order – Improving Critical Infrastructure Cybersecurity (2013); Presidential Policy Directive – Critical Infrastructure Security and Resilience 2013.

[93] Florian Skopik, Giuseppe Settanni, and Roman Fiedler, 'A problem shared is a problem halved: A survey on the dimensions of collective cyber defense through security information sharing' (2016) 60 Computers & Security 154.

On the business side, Sauerwein, Sillaber, and Breu found that phenomenon of 'shadow IT'—whereby day-to-day business operations are managed using IT products and services that are not explicitly approved or supported by the business—extends into the field of ISTI, with most professionals surveyed indicating that they regularly receive ISTI from sources that 'bypass security and risk management review processes or formal approval structures' and which is then often disseminated internally. A key motivation for this was identified as a concern for the timeliness of information, with 'almost half of the (shadow) cyber threat intelligence sources...accessed on a daily basis, or instantaneous'.[94]

Highlighting the value of including GL in our MLR, a number of reports and whitepapers exist detailing the field from a market perspective. Bromiley divides ISTI into the high-level categories of 'internal' and 'external', emphasising that 'true ISTI harmony exists when an organization uses both sources simultaneously' and providing a series of steps for making ISTI actionable:

- incorporating ISTI into an organization's security posture;
- using ISTI to help drive investigations and response;
- using ISTI to look into the past and possibly see things that were missing in the absence of the ISTI; and
- using ISTI to look into the future.[95]

Meanwhile, Brown and Lee found that the use of ISTI is growing, more organisations are consuming ISTI and 'information-sharing programs have value beyond just the information that is being shared.'[96] The findings echo those of Shackleford, who found that out of 326 qualified survey respondents, '69%...report[ed] implementing CTI to some extent, with only 16% saying they ha[d] no plans to pursue CTI in their environments.'[97] Brown and Lee also echo Sauerwein and others in suggesting there is a need for more automation, which they propose will allow 'organizations to better allocate resources and give analysts more time to focus on analysis and dissemination of intelligence, rather than on collection and processing of data.'

Finally, Falk categorises ISTI programmes in terms of maturity: aware; reactive; adaptive; purposeful; and strategic. The author also describes the idea of Threat Intelligence-as-a-Service (TIaaS).[98]

2.2.3 Market Research

Market research carried out by Gartner found that 'the term "threat intelligence" covers a diverse set of capabilities', and that this 'diversity of threat intelligence (TI) services, as well

[94] Clemens Sauerwein, Christian Sillaber, and Ruth Breu, 'Shadow cyber threat intelligence and its use in information security and risk management processes' [2018] Multikonferenz Wirtschaftsinformatik (MKWI 2018).

[95] Matt Bromiley, 'Threat intelligence: What it is, and how to use it effectively' [2016] SANS Institute InfoSec Reading Room.

[96] Rebekah Brown and Robert M Lee, 'The Evolution of Cyber Threat Intelligence (CTI): 2019 SANS CTI Survey' [2017] SANS Institute InfoSec Reading Room.

[97] Dave Shackleford, 'Who's using Cyberthreat Intelligence and how?' [2015] SANS Institute.

[98] Courtney Falk, Cyber Threat Intelligence as a Service (2017).

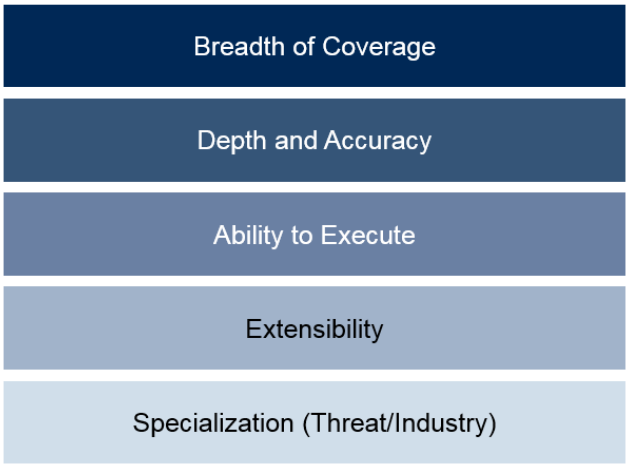


Figure 13: Methods to define and select an ISTI provider.[100]

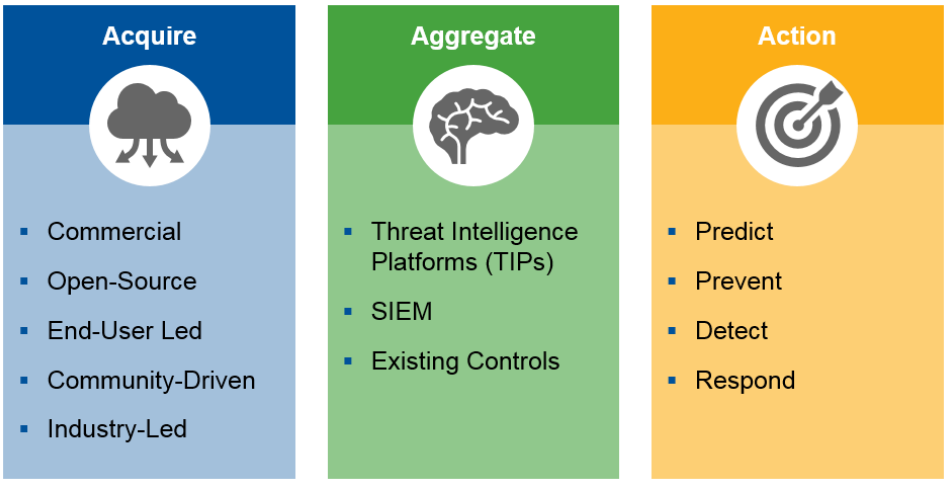


Figure 14: Three things to do well to achieve value from ISTI.[101]

as expertise, [has] created an environment in which purchasers often struggle to compare services, and there's still no single provider to address all of them.' The authors predict that '[b]y 1010, 20% of large enterprises will use commercial TI services to inform their security strategies', compared to 'fewer than 10% today.'[99]

The report authors define TI as follows:

> TI products and services provide knowledge about information security threats and other security-related issues. TI can provide information about the identities, motivations, characteristics and methods of threat actors.

However, they add that it is 'hard to pin down, in terms of a market definition,...what cleanly defines the types of solutions in this market.' TI can be both the main element of a product or service offering, or 'a feature of something larger' such as an enhancing element of a firewall or managed security service (MSS). These additional features, the authors write, 'should [be] treat[ed] as a differentiator for procuring and operating these products and services.'

[99] Craig Lawson, Ryan Benson, and Ruggero Contu, *Market Guide for Security Threat Intelligence Products and Services* (, Gartner 2019).

The authors of the report highlight two distinct approaches within the ISTI market:

- vendors who 'generate their own content, or, alternatively, provide what we consider to be original and substantial enrichment or aggregation of content harvested from other sources that has specialized analysis applied to it'; and
- vendors which a 'aggregate information from other sources and possibly add some metadata that they generate or is enriched by accessing other third parties', whose 'value is in stitching it all together to provide a cohesive outcome'.

They also distinguish between 'machine-readable TI (MRTI)' and 'TI geared at people'.

There is a shift underway in terms of who uses ISTI programs; whilst 'TI services appeal primarily to large enterprises that have significant brand presence or higher-risk profiles, and generally have security organizations with more-mature security programs...some service providers are expanding their focus to include mid-size organizations and have been pursuing this objective for several years by providing pre-packaged, easier-to-consume offerings at lower price points into a range of technologies and maturity levels.'

The report provides five methods that prospective ISTI product or service purchaser can use to determine the best option for their situation (see fig. 13). From the perspective of a service provider, the same methods can be used to choose which qualities to target with their offering, as we shall do later.

Finally, the report lists a number of providers and divides them into which of Gartner's three requirements for effective ISTI they most represent (see fig. 14):

- those that acquire intelligence (e.g., BAE Systems Applied Intelligence, CrowdStrike, etc.);
- those that aggregate intelligence (e.g., LookingGlass, ThreatConnect, etc.); and
- those that action intelligence (though the authors write that 'this is often embedded in [a] product or service itself').

2.2.4 Data Sources

There are many possible sources of data that an TIMS can call upon. However, more often than not they are heterogeneous and must be normalised if they are to be usefully processed; 'this normalization of disparate data sources is an extremely prosaic type of challenge, yet it is a sticking point that bungs up the works over and over again.'[103]

Data sources can be categorised along three dimensions: locus; structure; and trust.

Locus

A data source can be internal, such as the results of a penetration test or an internal intrusion detection service (IDS), or external, such as a public vulnerability database or a security researcher's blog. Whether a source is internal or external is unlikely to impact on whether it

[103] Henry Dalziel, *How to define and build an effective cyber threat intelligence capability* (Syngress 2014) p. 25.

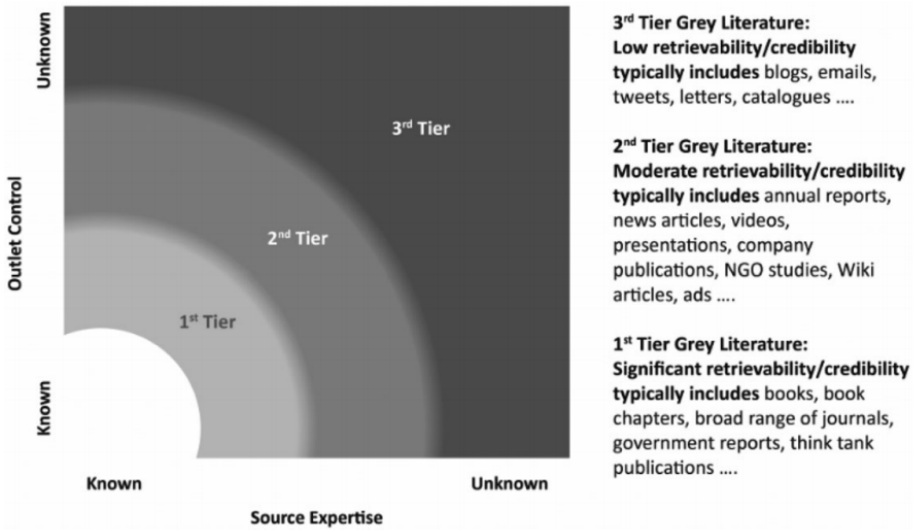


Figure 15: The shades of grey literature.[102]

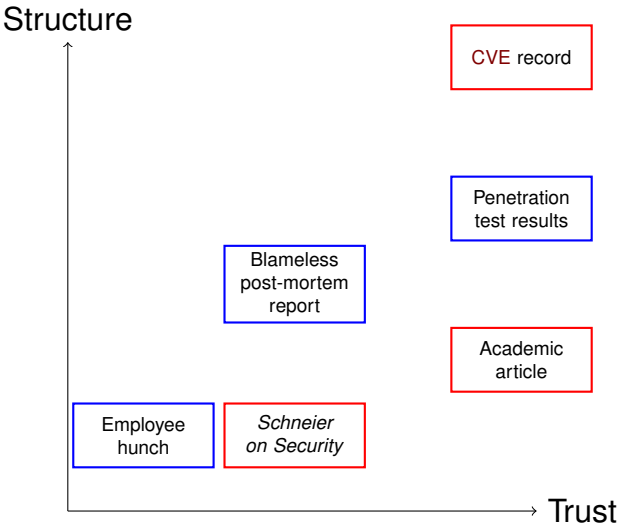


Figure 16: Categorisation of data sources.

is structured or unstructured, but it is very likely that internal sources will be inherently more trusted than external sources.

Structure

Some data sources are published in a structured format, such as the STIX™ for threat intelligence or the Common Vulnerabilities and Exposures (CVE) for vulnerabilities. Others are unstructured, such as blog posts and news articles. All data must be normalised before it can be usefully processed, and both structured and unstructured data can pose a problem here: unstructured data as it must have structure imposed on it, and structured data as it may need to be restructured if the TIMS uses a different approach.

Trust

Most important, perhaps, is the level of trust granted to a given data source. Fig. 15 shows the 'shades of grey' that can be applied when considering grey literature for inclusion in a multi-vocal literature review, and the same approach should be applied to ISTI data sources.

Less trusted sources should not necessarily be excluded, as they may still contain useful information that cannot be found elsewhere (as even an outright lie by an attacker can have criminological value[104]). However, this untrustworthiness should be made clear to all consumers of the data.

Trustworthiness can be affected by many factors: the perceived expertise of the source's author; a prior record of integrity; the presence of conflicts of interest; and so on. Note, however, that trust is not the same thing as certainty, which should also be made clear but relates to the perceived probability of an event occurring; see Kent.[105]

Fig. 16 shows the mapping of a number of data sources along these three axes: the level of trust increases left-to-right; the level of structure increases bottom-to-top; and the locus of the data is encoded as either red (for internal sources) or blue (for external). A truly comprehensive TIMS will have to utilise a wide range of different types of sources, but must consider the relative issues of each.

2.3 TIMS Reference Architecture

This section describes the reference architecture for a prospective threat intelligence management system (TIMS), based on our review of the academic literature, legislation and international standards.

2.3.1 TIMS Classification

A TIMS can be classified along a range of variables. We here propose the following:

- maturity;
- automation;
- source loci;
- sharing;
- interoperability; and
- specificity.

Maturity

As we have discussed, there are multiple competing proposals for modelling the intelligence cycle. However, we prefer the 'funnel of causality' suggested by Gill and Phythian (see § 2.2.2 and fig. 6). Marinos and Lourenço also offer a detailed map of ISTI programs, divided into strategic, tactical, technical and operational areas and the governance, collection, production and dissemination steps of the intelligence cycle (see fig. 17).

[104] Sveinung Sandberg, 'What can "lies" tell us about life? Notes towards a framework of narrative criminology' in *Advancing Qualitative Methods in Criminology and Criminal Justice* (Routledge 2014).
[105] Sherman Kent, 'Words of estimative probability' [1964] .

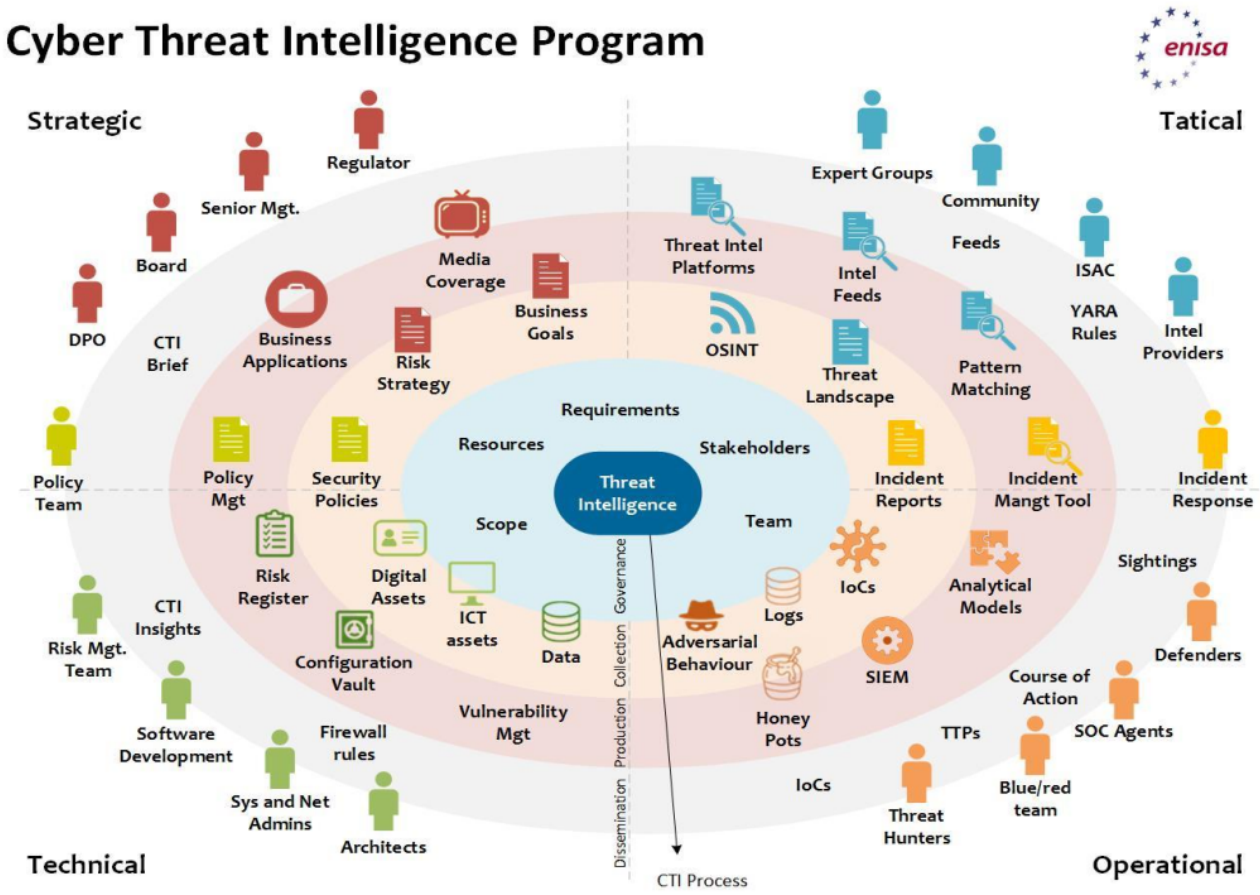


Figure 17: ISTI representation.[106]

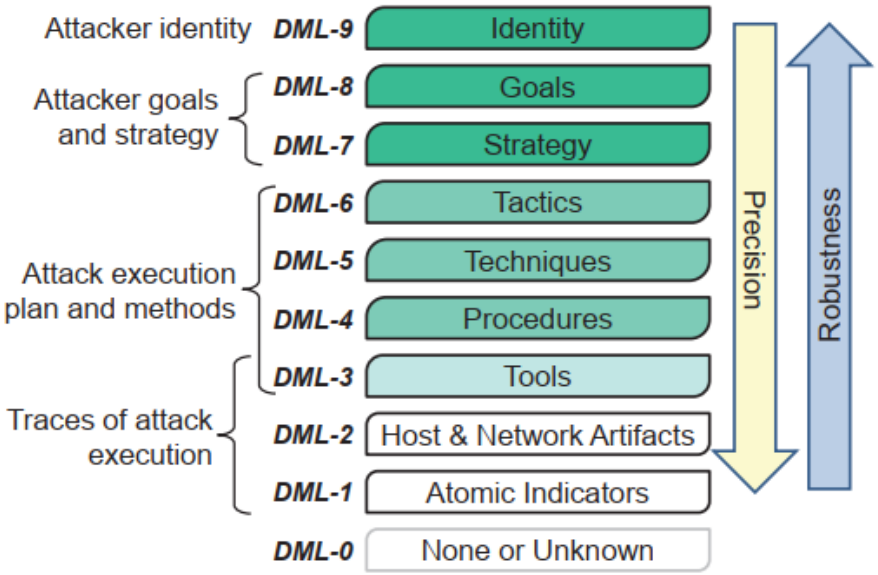


Figure 18: Detection Maturity Level model.[107]

Stillions proposed an eight-level Detection Maturity Level (DLM) model of ISTI solutions,[108] to which Bromander, Jøsang, and Eian added a ninth.[109] The levels of the model (see fig. 18) are as follows:

0. **None or Unknown**. There is no [Incident Response] team, or they are totally clueless.
1. **Atomic IoCs**. These are elementary pieces of host & network artifacts, which might have been received from other parties. The value of atomic IoCs is limited due to the short 'shelf life' of this type of information.
2. **Host & Network Artifacts**. This is the type of information which can be collected by network and endpoint sensors.
3. **Tools**. Attackers install and use tools within the victim's network...DML-3 means that the defender can reliably detect the attacker's tools, regardless of minor functionality changes to the tool...
4. **Procedures**. Detecting a procedure means detecting a sequence of two or more of the individual steps employed by the attacker.
5. **Techniques**. Techniques are specific ways of executing single steps of an attack.
6. **Tactics**. To detect a tactic means to understand how the attack has been designed and executed in terms [of] the techniques, procedures and tools used.
7. **Strategy**. This is a non-technical high-level description of the planned attack.
8. **Goals**. The motivation for the attack can be described as a goal.
9. **Identity**. The identity of the attacker, or the threat agent, can be the name of a person, an organisation or a nation state.[110]

Automation

Sauerwein and others and Brown and Lee both highlight the need for additional automation within available ISTI solutions.

NB: The remainder of this sub-section was not completed.

2.3.2 Architecture

Menges, Sperl, and Pernul attempt to unify the language and models used to describe ISTI into a single meta model (fig. 20), intended to provide 'a comprehensive specification, covering both the basic structuring elements and coherences that can be used to express intelligence information.'[113] Citing Burger and others, the authors state that 'basic CTI objects can be assigned to the three different categories Indicator, Intelligence and Attribution':[114]

[108] Ryan Stillions, 'The DML Model' (2014) ⟨https://ryanstillions.blogspot.com/2014/04/the-dml-model_21.html⟩ accessed 25 November 2019.
[109] Siri Bromander, Audun Jøsang, and Martin Eian, 'Semantic Cyberthreat Modelling.' (2016).
[110] ibid pp. 75–76.
[113] Florian Menges, Christine Sperl, and Günther Pernul, 'Unifying Cyber Threat Intelligence' (2019) p. 164.
[114] ibid p. 165.

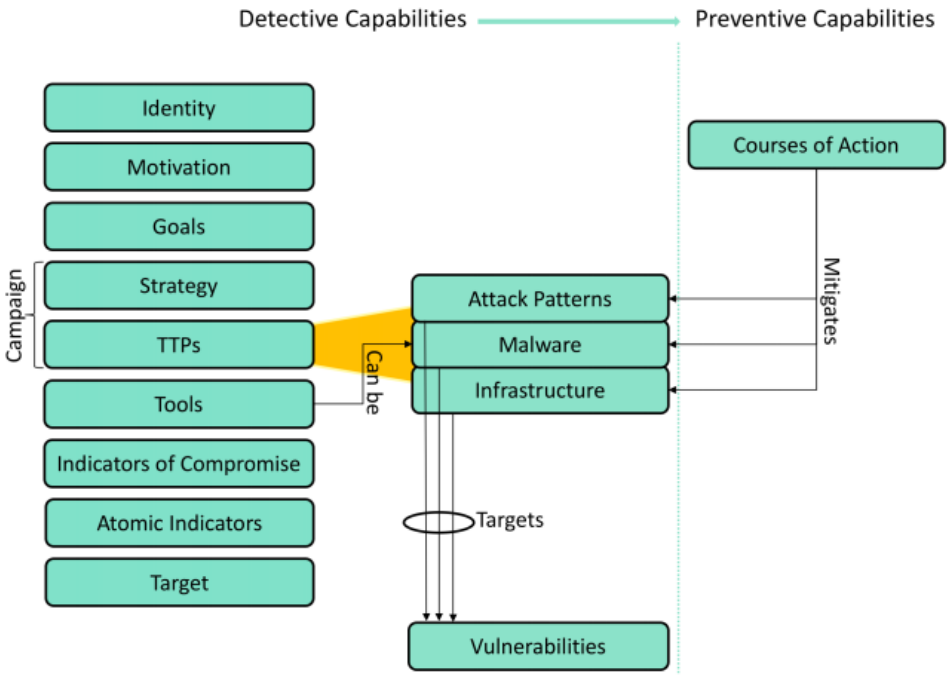


Figure 19: ISTI model.[111]

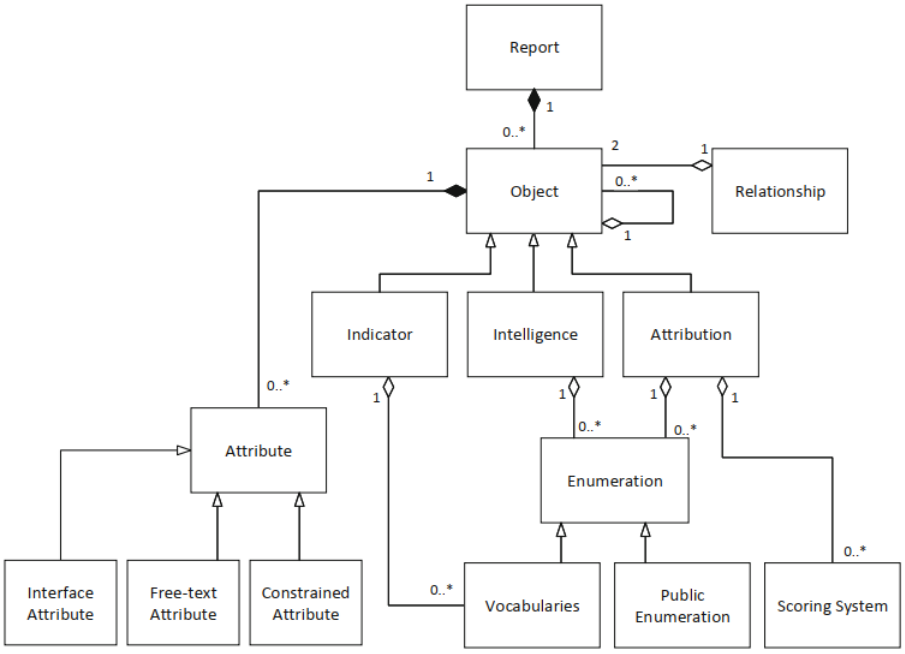


Figure 20: ISTI meta model.[112]

Component type	Classification	STIX	STIX2	IODEF 1&2	VERIS	X-ARF	Rule	Mapping
Indicator	Indicator	Indicator	Indicator	Indicator	Indicator	-	2	Indicator
Action	Indicator	Observable	Observed data	Record	Threat action	Attachment	3	Action
Attacker	Attribution	Threat actor	Threat actor	Threat actor	Actor	Source	1	Actor
Target	Attribution	Exploit target	Exploit target	System	Asset	Destination	1	Asset
Vulnerability	Attribution	Vulnerability	Vulnerability	Vulnerability	Vulnerability	Attachment	2	Vulnerability
Result	Attribution	Impact assessment	Impact assessment	Assessment	Impact assessment	-	1	Assessment
Campaign	Attribution	Campaign	Campaign	Campaign	Related incidents	-	2	Campaign
Method	Intelligence	TTP	Attack pattern	Attack pattern	Vector	Category	1	Attack pattern
Course of action	Intelligence	Course of action	Course of action	Defined COA	Corrective actions	-	2	Course of action
Incident	Intelligence	Incident	Report	Incident	Incident	Incident	2	Incident

Figure 21: Unified notation for threat intelligence.[118]

- 'Indicator objects describe patterns or behaviours that show the likelihood than an incident is occurring, has already occurred or will probably occur in the future';[115]
- 'Intelligence objects are used to represent specific knowledge about threats or incidents';[116] and
- 'Attribution objects describe the source, the target as well as the circumstances of an incident.'[117]

Following this, they construct a unified element notation by analysing the terms used by different standards and mapping these onto a single set of components, in three classifications (see fig. 21):

- Indicator:
  - Indicator; and
  - Action.
- Attribution:
  - Actor;
  - Asset;
  - Vulnerability;
  - (Impact) Assessment; and
  - Campaign.
- Intelligence:
  - Course of Action; and
  - Incident.

Finally, they use this model and notation to develop a unified data model for representing ISTI (fig. 22). This model divides the ten entities into three layers, aligned with the classifications presented in the unified notation, as well as including various entity relationships and values

[115] Menges, Sperl, and Pernul (n 113) p. 165.
[116] ibid p. 165.
[117] ibid p. 165.

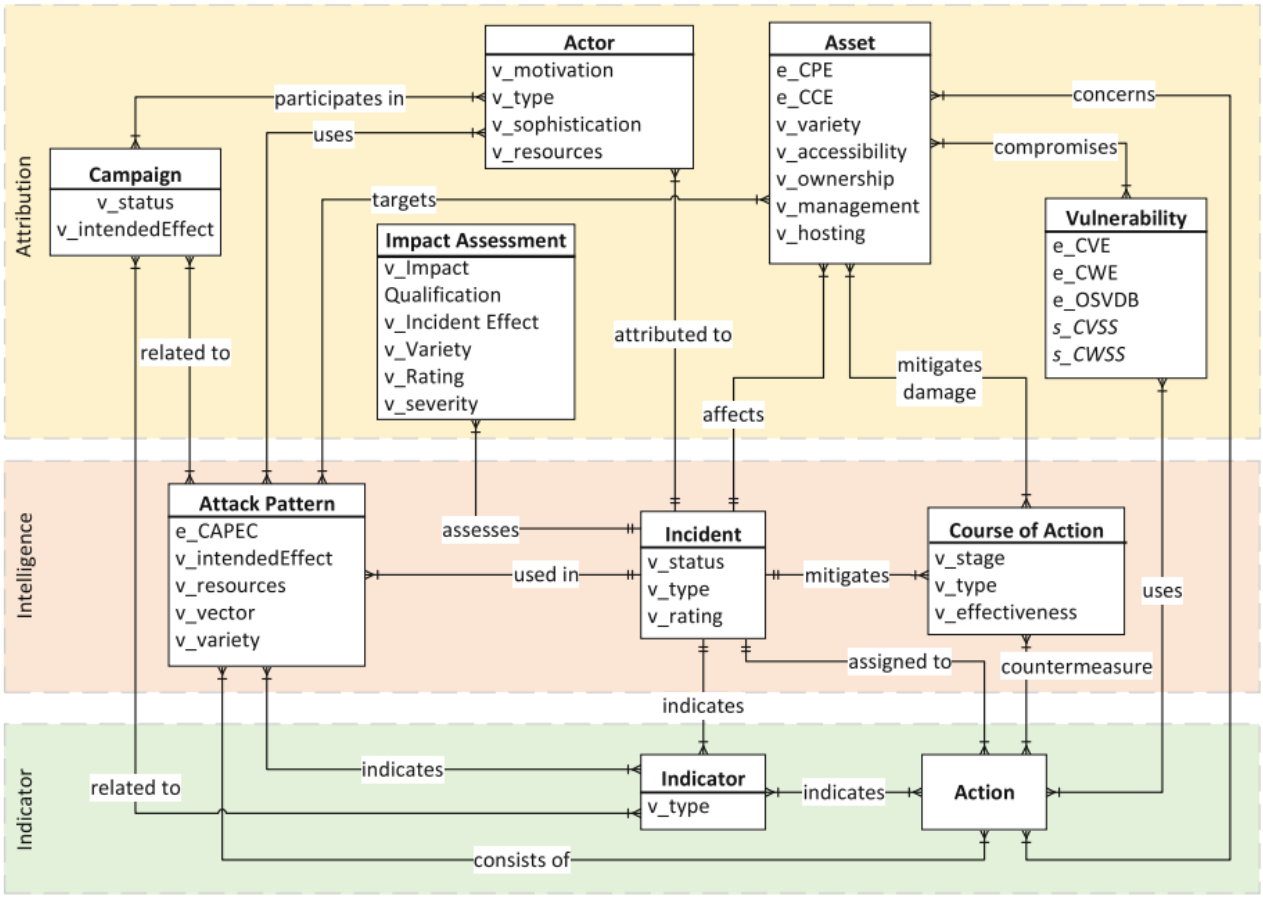


Figure 22: Unified ISTI data model.[121]

(where the ‘e_’ prefix denotes a public enumeration, ‘s_’ denotes a scoring system and ‘v_’ denotes a vocabulary or private enumeration). Menges, Sperl, and Pernul do not intend this model to be final, but rather aim to expose areas for future development; for example, they propose ‘[o]ne possible extension [to the *Asset* entity] would be an additional scoring system to evaluate the criticality of the assets’,[119] citing Kim and Kang.[120]

2.4 Software Development Practices

This section discusses contemporary software development practices. Beginning with an overview of the lean movement, we then describe various lightweight, iterative software development movements—Extreme Programming (XP), Agile, DevOps, etc.—and their relative pros and cons. We then examine the ISO/IEC/IEEE 12207 standard for software lifecycle process, and how it can be applied to agile approaches. We conclude by detailing the software development methodology to be used in the duration of this project.

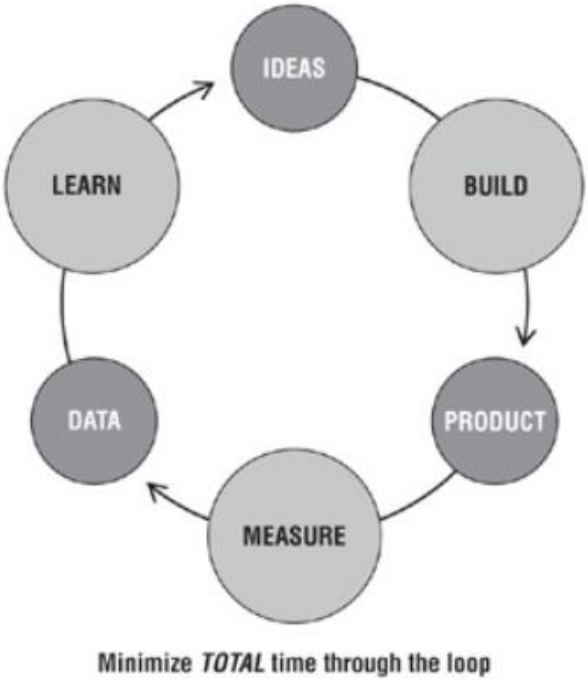


Figure 23: The Build-Measure-Learn feedback loop.[122]

2.4.1 Lean Thinking

'Lean thinking' was coined in 1996 to describe the ideas underpinning the greatly influential Toyota Production System.[123] The term 'lean' refers to the idea of an organisation or process stripped of all superfluity and healthier for it; critics accuse lean of prioritising cost-cutting above all other considerations, though this can perhaps more accurately be called 'emaciated thinking'.

Lean thinking considers a organisation or process in terms of:

- value;
- value streams (i.e., 'the process required to convert a business hypothesis into a technology-enabled service that delivers value to the customer'[124]); and
- flow.

The ultimate goal of a lean organisation or process is to minimise obstacles and friction to maximise the rate of flow through the defined value streams, with a subsequent increase in the value produced for customers. Note that in lean thinking, customers can be both internal and external to the organisation—a downstream operations team is considered a customer of an upstream software development team just as much as the customer who eventually hands over money for the finished product.

[119] Menges, Sperl, and Pernul (n **??**) p. 172.
[120] Anya Kim and Myong H Kang, *Determining asset criticality for cyber defense* (techspace rep, US Naval Research Laboratory 2011).
[123] James P Womack and Daniel T Jones, *Lean thinking: banish waste and create wealth in your corporation* (Second, first published 1996, Free Press 2003).
[124] Gene Kim and others, *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations* (IT Revolution 2016) p. 8.

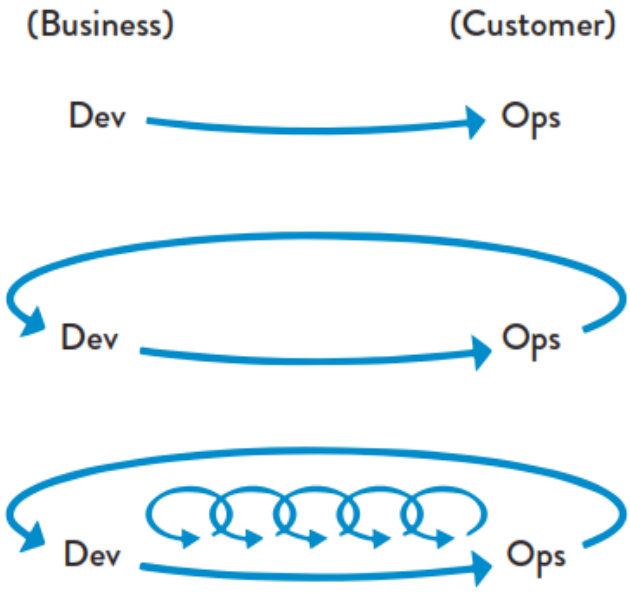


Figure 24: The Three Ways.[126]

Ries applies lean thinking to the running of a 'lean startup', with a 'startup' defined by the author as 'a human institution designed to create a new product or service under conditions of extreme uncertainty.'[125] Note that this definition need not refer to a startup company, and the book is replete with examples of startups operating within larger organisations.

The core tenets of the lean startup idea are rapid experimentation, regular reassessment and pervasive monitoring. 'Because startups often accidentally build something nobody wants, it doesn't matter if they do it on time and on budget', writes the author. 'The goal of a startup is to figure out the right thing to build—the thing customers want and will pay for—as quickly as possible.'[127] Key to this is the Build-Measure-Learn feedback loop (see fig 23) and the lean startup's goal of minimising the time taken to complete each cycle of the loop—this same idea is described by Kim as the 'Three Ways' (see fig. 24).[128]

1. flow;
2. feedback; and
3. continual learning and experimentation.[129]

The startup will have a vision—where it views itself after the success it believes to be possible. 'The first step would be to break down the grand vision into its component parts.'[130] This vision will, by virtue of the uncertainty inherent in being a startup, contain a number of leap-of-faith assumptions (e.g., 'customers will want to use this service', 'people who like the product will tell their friends', etc.), the two most important of which are the value hypothesis—'whether a product or service really delivers value to customers once they are

[125] Eric Ries, *The Lean Startup: How Constant Innovation Creates Radically Successful Businesses* (Portfolio Penguin 2011) p. 27.
[127] ibid p. 20.
[128] Gene Kim, The Three Ways: The Principles Underpinning DevOps, 'IT Revolution' (2016).
[129] Kim and others (n **??**) pp. 11–13.
[130] Ries (n 125) p. 61.

using it'—and the growth hypothesis—'how new customers will discover a product or service'. Vital at this early stage is to '...identify the elements of the plan that are assumptions rather than facts, and figure out ways to test them.'[131]

Ries describes three engines of growth that a startup may focus on:

- the sticky engine of growth, which relies on improving the customer retention rate;
- the viral engine of growth, which relies on increasing the number of new customers each customer introduces to the produce; and
- the paid engine of growth, which relies on increasing the margin between the value of a customer and the cost of acquiring them.[132]

Once the likely engine of growth and other hypotheses underpinning the vision are identified, the startup should build a Minimum Viable Product (MVP)—the most minimal product possible that can be used to validate or disprove the initial assumptions underlying the current strategy. Examples techniques include the concierge MVP in which the needs of early adopters are focused on in a way that would be impossible to scale and *Wizard of Oz* testing in which customers believe they are interacting with an automated system that is in fact human operated. This allows them to rapidly move from the Build phase of the feedback loop to the Measure phase.

The MVP will be a far cry from the startup's vision, but this is by design. 'Failure is a prerequisite to learning. The problem with the notion of shipping a product and then seeing what happens is that you are guaranteed to succeed—at seeing what happens. But then what? As soon as you have a handful of customers, you're likely to have five opinions about what to do next. Which should you listen to?'[133] Between receiving customer feedback from early adopters and monitoring various metrics that will allow them to validate or disprove their hypotheses, the startup can receive rapid feedback on each change they make to the product and identify cause and effect, focusing their efforts on those changes that tune their chosen engine of growth and leaving those that do not.

When the startup reaches the point in which they have exhausted the product changes they can use to tune their engine of growth, they must decide whether to pivot or persevere. They may have found through testing that their initial hypotheses were incorrect—perhaps their customers, though satisfied with the product, do not tell their friends as expected, in which case the viral engine of growth may not be appropriate. Or they may have made progress from their initial MVP, but not as fast as they believe they should be. Perhaps they have experienced substantial progress, but are afraid they have only achieved a local maxima and that greater progress is available elsewhere.

Ries identifies various different types of potential pivot:

- the zoom-in pivot, in which a sub-feature of the product becomes the new focus;
- the zoom-out pivot, in which the product becomes a sub-feature of a larger product;
- the customer segment pivot;

[131] Ries (n 125) p. 69.
[132] ibid pp. 209–219.
[133] ibid p. 154.

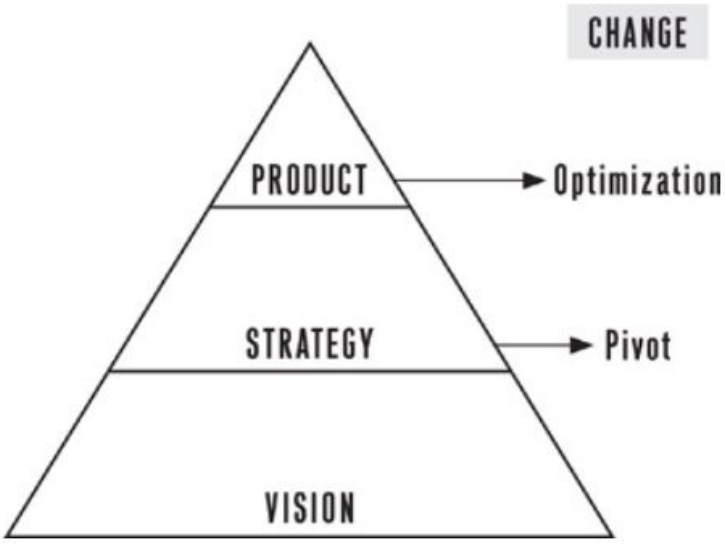


Figure 25: The Vision-Strategy Product pyramid.[134]

- the customer need pivot, in which the need the product aims to satisfy is changed;
- the platform pivot, shifting from an application to a platform or vice versa;
- the business architecture pivot, shifting from high volume-low margin to low volume-high margin or vice versa;
- the value capture pivot, changing the way the startup captures the value they create (e.g., their revenue model);
- the engine of growth pivot;
- the channel pivot; and
- the technology pivot.[135]

'The sign of a successful pivot is that these engine-tuning activities are more productive after the pivot than before.'[136] The theory is that by rapidly optimising the product and occasionally pivoting the strategy, the startup will converge more rapidly on a product that fulfils their original vision (see fig. 25).

2.4.2 Extreme Programming

Extreme Programming (XP) is a software development methodology primarily developed by Beck in the late 1990s and early 2000s. The 'Extreme' in the title refers to the goal of the method, which is to take those elements of software development commonly viewed as beneficial to the extreme.

Similar to the lean movement, the focus of XP is to minimise the time taken for developers to receive feedback on their code. XP endorses five key values:

[135] Ries (n **??**) pp. 172–176.
[136] ibid p. 118.

- communication;
- feedback;
- respect.
- simplicity; and
- courage;

These values are underpinned by various rules, which are designed to be interdependent and which serve to codify some aspects of lean thinking. Different sets of rules exist, but some examples include:

- a stand up meeting starts each day;
- code the unit test first;
- never add functionality early;
- the customer is always available; and
- integrate often.[137]

Finally, XP is implemented through twelve practices, grouped into four areas:

1. Fine-scale feedback:
   (a) pair programming;
   (b) planning game;
   (c) test-driven development; and
   (d) whole team.
2. Continuous process:
   (a) continuous integration, or constantly merging new code into the main branch;
   (b) refactoring, or going back to previous work to improve it; and
   (c) small releases.
3. Shared understanding:
   (a) coding standards;
   (b) collective code ownership;
   (c) simple design; and
   (d) system metaphor.
4. Programmer welfare:
   (a) sustainable pace.[138]

Whilst XP was influential at the turn of the century, it was criticised in some quarters for being overly prescriptive and requiring unrealistic levels of discipline from implementers.

[137] Don Wells, The Rules of Extreme Programming (1999).
[138] Kent Beck, *Extreme Programming Explained: Embrace Change* (Second, first published 1999, Addison-Wesley Professional 2004).

2.4.3 Agile

The *Manifesto for Agile Software Development* was issued in 2001 and signed by a multiple influential software developers, including Beck.[139] The manifesto identified four dichotomies within software development and expressed the signatories' belief that 'while there is value in the items on the right we value the items on the left more'; the items are:

- Individuals and Interactions over processes and tools;
- Working Software over comprehensive documentation;
- Customer Collaboration over contract negotiation; and
- Responding to Change over following a plan.[140]

Similarly to XP, the *Manifesto* also codified twelve principles:

1. satisfy the customer through early and continuous delivery of valuable software;
2. welcome changing requirements, even in late development;
3. deliver working software frequently;
4. business people and developers must work together daily throughout the project;
5. build projects around motivated individuals...and trust them to get the job done;
6. the most efficient and effective method of conveying information...is face-to-face conversation;
7. working software is the primary measure of progress;
8. the sponsors, developers, and users should be able to maintain a constant pace indefinitely;
9. continuous attention to technical excellence and good design;
10. simplicity—the art of maximizing the amount of work not done—is essential;
11. the best architectures, requirements, and designs emerge from self-organizing teams; and
12. at regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.

Agile is often mistaken as a specific method of software development, rather than an overarching philosophy that is realised through various methods (e.g., Scrum). A useful metaphor is provided by Littlefield:

[139] Kent Beck and others, Manifesto for Agile Software Development (2001).
[140] ibid.

> A good analogy would be the difference between a recipe and a diet. A vegetarian diet is a set of methods and practices based on principles and values. A recipe for chickpea tacos would be a framework you can use to implement your vegetarian diet.
>
> This is similar to the relationship between Agile (the diet) and Scrum (the recipe you follow).[141]

The complete list of methods and frameworks considered to be Agile is too long to list here. Instead, we shall now go through each principle of the *Manifesto* and explain how it can be adhered to within a project, highlighting along the way a few popular methods and frameworks.

1. Satisfy the customer through early and continuous delivery of valuable software.

As many of the processes standing between a developer writing code and a customer receiving it should be automated as possible. There are three levels of this automation:

1. continuous integration, in which developers commit new code to the master code branch regularly (at least once per day);
2. continuous delivery, in which code is shown to be a deployable at all times, but deployment to production is still manual; and
3. continuous deployment, in which the deployment to production is automated.

Obviously, successfully implementing continuous methods requires discipline from developers and comprehensive automated testing suites. One cannot jump to continuous delivery or deployment in one go.

However, one can begin with something like do-nothing scripting, which will highlight areas of potential automation along with delivering immediate benefits in the form of process standardisation and error avoidance. Gradually, these processes can become more and more automated.[142]

At the same time, continuous integration can and should be implemented as soon as possible. Developers having to regularly commit their code to master encourages them to shift into working in smaller, more incremental chunks.

2. Welcome changing requirements, even in late development

In order to ensure the the project is reactive to changing requirements, development must be broken into small intervals. At the end of each interval, the results should be assessed and customers consulted to ensure that any requirement changes can be incorporated into planning at the earliest possible stage. One such method of iterative development is Scrum.

Scrum originated in a 1995 paper as a project management framework built around small teams breaking their work down into short iterations (called 'sprints'), tracking their progress and feeding the results back into the planning of future iterations.[143] The authors of that paper now maintain *The Scrum Guide*, a *de facto* official guide to the method.[144]

[141] Andrew Littlefield, The Beginner's Guide to Scrum and Agile Project Management (2016).

[142] Dan Slimmon, Do-nothing scripting: the key to gradual automation (2019).

[143] Ken Schwaber, 'SCRUM Development Process' (Cory Sutherland Jeff and Casanave and others eds, Springer London 1997).

[144] Jeff Sutherland and Ken Schwaber, *The Scrum Guide™: The Definitive Guide to Scrum: The Rules of the Game* (2017).

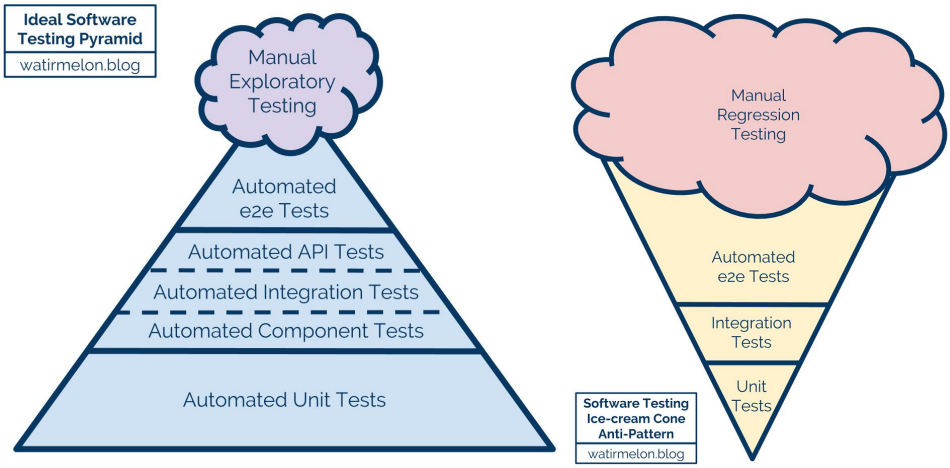


Figure 26: Ideal and non-ideal testing pyramids.[146]

The Scrum framework defines three roles:

- the product owner, who represents the product's stakeholders, has final say on backlog prioritisation and is ultimately responsible for project results;
- the development team, ideally formed of three to nine members who decide upon the technical solution to the business problems posed by the product owner and build features during sprints; and
- the Scrum Master, who removes obstacles to delivery and facilitates the Scrum process.

When a new feature proposal arises, the product owner will write a customer-focused ticket (e.g., by creating a user story) and add it to the product backlog. When a sprint is planned, a goal is set and relevant items moved from the product backlog to the sprint backlog. A sprint should ideally be between one week and one month long, with the emphasis on valuable, useful output—see the below definition of 'done':

> At the end of each development interval, we must have integrated, tested, working, and potentially shippable code, demonstrated in a production-like environment, created from trunk using a one-click process, and validated with automated tests.[145]

At the end of the pre-assigned duration for the sprint, the team finish their work regardless of whether the goal has been achieved or not. A sprint is held in which the team review the work they have completed and not completed, present the work to stakeholders and begin planning the next sprint. This is followed by a sprint, in which the team reflect on the previous sprint and identify areas where processes could be improved. For example, if the sprint completed with many unfinished tasks, the team may take this into account by being more realistic in the planning of their next sprint.

3. Deliver working software frequently.

[145]Kim and others (n **??**) p.149.

In Scrum, only features that can be shown to be working—i.e., 'done', as per the definition above—can be shown to stakeholders at the sprint. For continuous delivery or deployment to be possible, comprehensive automated test suites are needed. These tests should be performed in ascending order of time taken to complete—quick automated unit tests first, followed by slower automated component, integration and Application Programming Interface (API) testing, followed by even slower automated Graphical User Interface (GUI) tests and only reaching the final, labour-intensive round of manual testing if the build has passed all previous stages (see fig. 26).[147]

One way of ensuring comprehensive test coverage is to employ Test-Driven Development (TDD). This can be summed up in three steps:

1. Ensure the tests fail. Write a test for the next bit of functionality you want to add. Check in.
2. Ensure the tests pass. Write the functional code until the test passes. Check in.
3. Refactor both new and old code to make it well structured. Ensure the tests pass. Check in again.[148]

Nagappan and others found that for teams using TDD 'the pre-release defect density of the four products decreased between 40% and 90% relative to similar projects that did not use the TDD practice', with the teams experiencing only '15–35% increase in initial development time after adopting TDD'.[149]

4. Business people and developers must work together daily throughout the project.

The DevOps (and, subsequently, DevSecOps) movement is focused on breaking down barriers to communication between different groups within your organisation in the interest of reducing siloisation and promoting skill transfer and cross-communication. The movement originally focused on harmonising the daily work of development and operations teams, the latter of which could be considered 'business people'. DevOps is not, as Kim and others stress, about 'the complete elimination of the IT Operations function'; rather, it is about 'enabl[ing] developer productivity through APIs and self-serviced platforms that create environments, test and deploy code, monitor and display production telemetry, and so forth' with the result that 'IT Operations become more like Development (as do QA and Infosec), engaged in product development, where the product is the platform that developers use to safely, quickly, and securely test, deploy and run their IT services in production.'[150]

One advancement that has helped enable these developments is the shift to Infrastructure as Code (IaC), where configuration settings for product infrastructure are stored in machine-readable formats (e.g., YAML files) and checked into version control alongside the product code. This ensures that environments can be easily spun up and down on-demand by developers, settings consistency can be guaranteed across multiple instances and IT Operations gain experience in using the same tools as developers. Similarly, some organisations (e.g., Google) require developers to self-manage their production services until it operates within certain non-functional parameters before it can be passed on to the IT Operations team, who

[147] Kim and others (n **??**) pp. 123–141.
[148] ibid pp. 134–135.
[149] Nachiappan Nagappan and others, 'Realizing quality improvement through test driven development: results and experiences of four industrial teams' (2008) 13 Empirical Software Engineering 289.
[150] Kim and others (n **??**) p. xvi.

reserve the right to hand it back to the developers in future should performance degrade significantly.[151]

5. Build projects around motivated individuals...and trust them to get the job done.

Kim and others write that one way '...to enable high-performing outcomes is to create stable service teams with ongoing funding to execute their own strategy and road map of initiatives.' They advocate 'fund[ing] not projects, but services and products', with the goal of 'valu[ing] the achievement of organizational and customer outcomes, such as revenue, customer lifetime value, or customer adoption rate, ideally with the minimum of output'.[152]

The point about trusting employees to get the job done brings up the question of working practices. Despite many organisations having policies on remote working and the like, these are often treated in practice as more of a privilege that is granted rather than a right that is exercised. There is substantial evidence in favour of remote working, suggesting benefits ranging from productivity increases, improved workforce diversity, increased employee retention, better employee mental health and supporting family relationships.[153]

6. The most efficient and effective method of conveying information...is face-to-face conversation.

'As organizations grow, one of the largest challenges is maintaining effective communication and coordination between people and teams...Collaboration is...impeded...when teams are separated by contractual boundaries, such as when work is performed by an outsourced team.'[154] Note, however, that 'face-to-face' does not necessarily have to mean in-person—instant messaging and video conferencing are workable substitutes. One tool commonly deployed in support of this Agile principle is the daily stand-up meeting in which all developers start the day by briefly telling the team what they worked on yesterday, what they will work on today and any obstacles they expect to encounter. Insisting that participants stand up is intended to help keep the meetings short, as is the prohibition of longer discussions—where these are necessary, those who need to discuss are expected to arrange a separate meet-up between themselves.

7. Working software is the primary measure of progress.

This principle captures the idea that judging project success by how on-time or within-budget it currently is does not provide useful information as to whether the idea itself is sound. It is not possible to perform user research without working software, and 'if we are not performing user research, the odds are that two-thirds of the features we are building deliver *zero* or *negative* value to our organisation, even as they make our codebase ever more complex, thus increasing maintenance costs over time and making our software more difficult to change. Furthermore, the effort to build these features is often made at the expense of delivering features that *would* deliver value (i.e., opportunity cost).'[155]

8. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Agile is opposed to the late nights, enforced overtime and end-of-project 'crunch time' that

[151] Kim and others (n **??**) p. 234–240.

[152] ibid p. 87.

[153] Nicholas Bloom and others, 'Does working from home work? Evidence from a Chinese experiment' (2014) 130(1) The Quarterly Journal of Economics 165; Owl Labs, *State of Remote Working 2019* (, Owl Labs 2019); Jobsite, The rise of women in technology (2017); Adrianne Bibby, 17 Stats About Remote Work in 2019 (2019).

[154] Kim and others (n **??**) p. 88.

[155] ibid p. 244.

characterise many other software development projects, perhaps most notoriously within the computer game industry.[156] It aspires to creating a pace of development that, whilst still challenging the team, is sustainable in the long-term.

Kanban is one method of balancing customer demand and team capacity in project management. Tasks to complete are visualised, most commonly via sticky notes on a whiteboard that is visible at all times. The whiteboard is divided into buckets; for example, 'backlog', 'working on', 'testing', 'deployment' and 'delivered'. Tasks are moved across the board as they complete each stage, making their way towards the final state on the right-hand side. This allows the team and other stakeholders to see, at a glance, the progress of different features and the current capacity of the team to take on additional tasks.

An important feature of Kanban is the maximum limit for each bucket. Once a bucket is full of tasks, no new ones can be assigned until space is cleared up by moving existing tasks to later buckets. This both ensures that the team are never operating above capacity and helps to quickly identify potential bottlenecks, where process improvements can be focused. For example, if the 'deployment' bucket is routinely holding up tasks, it may be that too many levels of approval are required or the process could be further automated.

9. Continuous attention to technical excellence and good design.

As previously mentioned, a vital part of lean thinking is measuring and learning. Telemetry should be collected for all features—'if it was important enough for an engineer to implement, it is certainly important enough to generate enough production telemetry so that we can confirm that it is operating as designed and that the desired outcomes are being achieved.'[157] Detailed user metrics should be recorded to allow for more valuable forms of analysis—cohort analysis is a far more powerful tool than simple gross or cumulative numbers, for example.

All metrics must satisfy the three 'A's:

- Actionable (i.e., must demonstrate clear cause and effect);
- Accessible (i.e., everyone must be able to access them from a centralised repository); and
- Auditable (i.e., must be backed up with real-world tests to ensure correctness).

For example, cohort testing can show how many of the visitors to a produce Web site in a given month ultimately convert into paying customers (e.g., the sales funnel). This can then be used to compare the results of an A/B test of two different pricing strategies or Web site layouts to establish a cause and effect relationship.

Modern monitoring architecture must feature two elements:

- data collection at the business logic, application, and environments layer; and
- an event router responsible for storing our events and metrics.[158]

[156] Tim Surette, EA settles OT dispute, disgruntled "spouse" outed, 'GameSpot' (2006); Rockstar Spouse, Wives of Rockstar San Diego employees have collected themselves, 'Gamasutra' ; Jason Schreier, Inside Rockstar Games' Culture Of Crunch, 'Kotaku' .

[157] Kim and others (n **??**) p. 201.

[158] ibid p. 199.

The benefits of pervasive monitoring go beyond just fault detection. For example, Jacobson, Yuan, and Joshi presents the example of Netflix's predictive traffic scaling engine, which identifies usage patterns in previously-recorded traffic data and scaling their servers up and down appropriately.[159]

An important Agile concept is the information radiator—any number of highly-visible displays that allow everybody to see at a glance any key metrics about a product, such as uptime or usage patterns. 'We want to make as much infrastructure telemetry visible as possible, across all the technology stakeholders, ideally organized by service or application. In order words, when something goes wrong with something in our environment, we need to know exactly what applications and services could be or are being affected.'[160] This should also include security metrics: 'by radiating how our services are being attacked in the production environment, we reinforce that everyone needs to be thinking about security risks and designing countermeasures in their daily work.'[161]

10. Simplicity—the art of maximizing the amount of work not done—is essential.

In its most obvious incarnation, this principle drives the mentality within Agile of 'automate all the things'. Any process that can be automated, should be. This can apply to testing, to deployment, to monitoring, to alerting, etc. The principle also underlines the notion of the MVP and rapid experimentation within lean startup theory—work performed on the basis of an untested assumption should be minimised, and additional time and resources should only be committed once the initial assumption(s) seem to be validated.

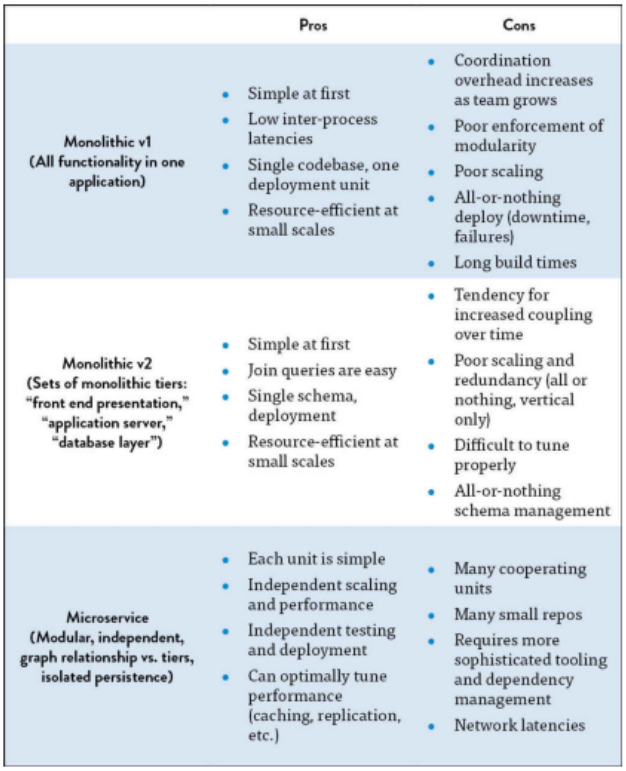
11. The best architectures, requirements, and designs emerge from self-organizing teams.

Team structure can have an impact on their resulting software; Conway's law states that 'organizations which design systems...are constrained to produce designs which are copies of the communication structures of these organizations.'[163] Teams should ideally be provided with the autonomy to determine their own priorities and plans: '[a] way to enable high-performing outcomes is to create stable service teams with ongoing funding to execute their own strategy and road map of initiatives [and] the dedicated engineers needed to deliver on concrete commitments made to internal and external customers, such as features, stories and tasks.'[164]

On the topic of architectures, Shoup describes three archetypes, which are summarised in fig. 27.[165] Micro-services are also referred to as Service-Oriented Architectures (SOAs): 'The idea is that developers should be able to understand and update the code of a service without knowing anything about the internals of its peer services. Services interact with their peers strictly through APIs and thus don't share data structures, database schemata, or other internal representations of objects.'[166]

12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.

[159] Daniel Jacobson, Danny Yuan, and Neeraj Joshi, Scryer: Netflix's predictive auto scaling engine, 'Netflix Technology Blog' (2013).
[160] Kim and others (n **??**) p. 212.
[161] ibid p. 327.
[163] Melvin E Conway, 'How do committees invent' (1968) 14(4) Datamation 28.
[164] Kim and others (n **??**) p. 87.
[165] Randy Shoup, 'From the Monolith to Microservices: Lessons from Google and eBay' (2016) vol 2016.
[166] Kim and others (n **??**) p. 89.



	Pros	Cons
Monolithic v1 (All functionality in one application)	• Simple at first • Low inter-process latencies • Single codebase, one deployment unit • Resource-efficient at small scales	• Coordination overhead increases as team grows • Poor enforcement of modularity • Poor scaling • All-or-nothing deploy (downtime, failures) • Long build times
Monolithic v2 (Sets of monolithic tiers: "front end presentation," "application server," "database layer")	• Simple at first • Join queries are easy • Single schema, deployment • Resource-efficient at small scales	• Tendency for increased coupling over time • Poor scaling and redundancy (all or nothing, vertical only) • Difficult to tune properly • All-or-nothing schema management
Microservice (Modular, independent, graph relationship vs. tiers, isolated persistence)	• Each unit is simple • Independent scaling and performance • Independent testing and deployment • Can optimally tune performance (caching, replication, etc.)	• Many cooperating units • Many small repos • Requires more sophisticated tooling and dependency management • Network latencies

Figure 27: Architectural archetypes.[162]

The notion of technical debt, or the gradual accrual of workarounds, good-enoughs and bodge jobs that are the avoidable side-effect of any software development is important to lean thinking. Like financial debt, technical debt must be regularly paid down with scheduled improvement blitzes—periods of time dedicated solely to the improvement of processes and workarounds. A common figure is that this should amount to 20% of the development time (i.e., every fifth sprint should be an improvement blitz).[167]

On a larger scale, Dekker describes the idea of a 'just culture' as one that accepts that 'human error is not our cause of troubles; instead, human error is a consequence of the design of the tools we gave them'.[168] Kim and others describe two practices that can help to create a just culture:

- blameless post-mortems; and
- the controlled introduction of failures into production.

The blameless post-mortem should occur following a significant incident such as a customer-affecting bug or a deployment failure, as soon as possible once the issue has been resolved in order to ensure memories are fresh and cause and effect can be established. The steps are as follows:

1. construct a timeline;
2. empower all engineers to improve safety by allowing them to give detailed accounts of their contributions to failures;

[167] Kim and others (n **??**) pp. 299–303.
[168] Sidney Dekker, *Just Culture: Balancing safety and accountability* (CRC Press 2016).

3. enable and encourage people who do make mistakes to be the experts who educate the rest of the organisation on how not to make them in future;
4. accept that there is always a discretionary space where humans can decide to take action or not, and that the judgement of those decisions lies in hindsight; and
5. propose countermeasures to prevent a similar accident from happening in the future.[169]

Attending the post-mortem should be everyone involved in the decisions that contributed to the incident, those who identified, responded to, diagnosed and were affected by the problem and anyone else who is interested. The timeline should be supported with evidence (e.g., emails, chat logs) where possible, and the post mortem report should be offered to the rest of the organisation in an accessible place in the event of similar issues in future. The blamelessness is vital: 'when engineers make mistakes and feel safe when giving details about it, they are not only willing to be held accountable, but they are also enthusiastic in helping the rest of the company avoid the same error in the future. This is what creates organizational learning.'[170]

Controlled failure introduction, on the other hand, ensures that issues that are at risk of happening at some point (e.g., server loss) do not have to be waited on, but rather simulated in a controlled manner and lessons learned immediately. Kim and others cite a number of examples, such as Netflix's Chaos Monkey program which randomly kills processes running in production and the use of pre-planned Game Days to simulate a major issue (be it an Amazon Web Services (AWS) outage or an alien invasion).

2.4.4 ISO/IEC/IEEE 12207

The ISO/IEC/IEEE 12207 standard was first introduced in 1995, and was most recently revised in 2017.[172] Prior to this revision it was known as just ISO/IEC 12207.

The standard presents a set of processes 'from which an organization can construct software life cycle models appropriate to its products and services.'[173] As such, it is intended to be framework-, life cycle model- and technique-agnostic.

The standard divides its thirty processes into four process groups (see fig. 28):

1. Agreement processes:
   (a) Acquisition process; and
   (b) Supply process.
2. Organizational Project-enabling processes:
   (a) Life Cycle Model Management process;
   (b) Infrastructure Management process;
   (c) Portfolio Management process;

[169] Kim and others (n **??**) pp. 274–275.
[170] ibid p. 274.
[172] ISO/IEC/IEEE 12207:2017: Systems and software engineering — Software life cycle processes (2017).
[173] ibid p. vii.

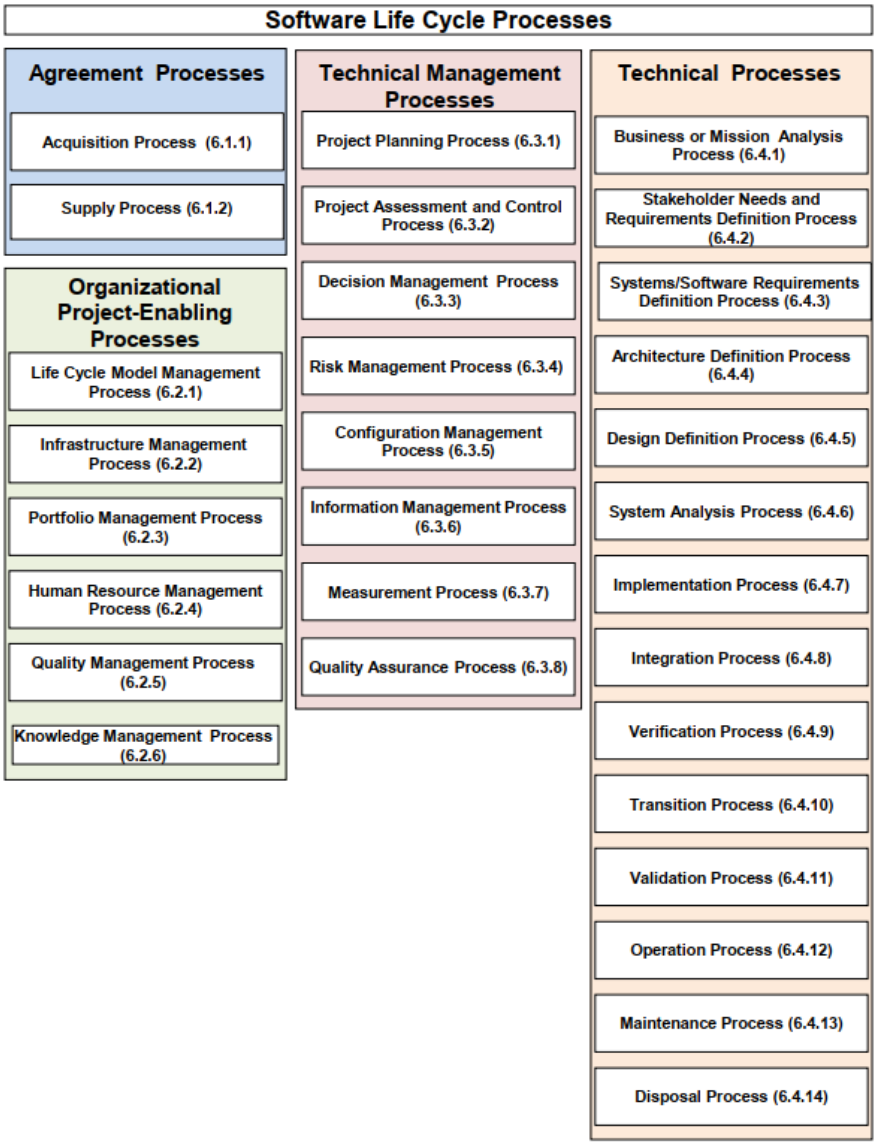


Figure 28: ISO/IEC/IEEE 12207 software life cycle processes.[171]

(d) Human Resource Management process;
(e) Quality Management process; and
(f) Knowledge Management process.

3. Technical Management processes:

(a) Project Planning process;
(b) Project Assessment and Control process;
(c) Decision Management process;
(d) Risk Management process;
(e) Configuration Management process;
(f) Information Management process;
(g) Measurement process; and
(h) Quality Assurance process.

4. Technical processes:

(a) Business of Mission Analysis process;
(b) Stakeholder Needs and Requirements Definition process;
(c) Systems/Software Requirements Definition process;
(d) Architecture Definition process;
(e) Design Definition process;
(f) System Analysis process;
(g) Implementation process;
(h) Integration process;
(i) Verification process;
(j) Transition process;
(k) Validation process;
(l) Operation process;
(m) Maintenance process; and
(n) Disposal process.

We shall now provide a brief overview of each process group.

Agreement Processes

The standard states that '[t]he Agreement processes are organizational processes that apply outside of the span of a project's lifespan, as well as for a project's lifespan.'[174] They can be considered meta-processes, which set the terms of engagement between producers and users of a system and lead to its creation.

Organisational Project-Enabling processes

[174] ISO/IEC/IEEE 12207:2017 (n 172) p. 21.

These processes 'are concerned with providing the resources to enable the project to meet the needs and expectations of the organization's stakeholders.' They 'establish the environment in which projects are conducted.'[175]

Technical Management processes

Processes within this group 'are concerned with managing the resources and assets allocated by organization management and with applying them to fulfil...agreements.'[176] Note that, again, these processes cover the full life cycle of the software, from planning to quality assurance (which, depending on the development methodology used, my take place at the start and finish of the project, respectively, or continuously throughout).

Technical processes

These are the processes that 'transform the needs of stakeholders into a product or service.' In short, they bring the goods. Unlike the other process groups, '[t]he Technical processes are applied in order to create and use a software system.'[177]

As previously mentioned, the standard is intended to be methodology-agnostic. As such, there is no problem integrating it with agile methodologies such as Scrum, as discussed in Annex H of the standard and by Irrazabal and others (although there in terms of the ISO/IEC 12207:2008 revision of the standard).[178] Though 'the life cycle models used in agile projects are often highly incremental and evolutionary...organizations that use agile methods do apply the life cycle processes identified in this document'.[179]

Appendix B details how each process will be implemented for this project.

2.4.5 Proposed Development Methodology

The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL', 'SHALL NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'MAY', and 'OPTIONAL' in this document are to be interpreted as described in RFC 2119.[180]

For the remainder of this Knowledge Transfer Partnership (KTP), development shall adhere to the Agile philosophy, implemented in practice through use of the following techniques and rules:

- Scrum SHALL be used for project planning and time management.
  - Sprints SHOULD be two weeks long.
  - Occasionally longer or shorter sprints MAY be needed, but these MUST be no longer than one month and no shorter than one week.
  - If the number of longer or shorter sprints reaches 20 % of overall sprints, the default timeframe SHOULD be reviewed.
  - There SHOULD be one (and only one) Product Owner per project who has ultimate decision-making authority for that project.

[175] ISO/IEC/IEEE 12207:2017 (n 172) p. 22.
[176] ibid p. 22.
[177] ibid p. 22.
[178] Emanuel Irrazabal and others, 'Applying ISO/IEC 12207: 2008 with SCRUM and Agile methods' (2011).
[179] ISO/IEC/IEEE 12207:2017 (n 172) p. 127.
[180] Scott Bradner, Key words for use in RFCs to Indicate Requirement Levels (1997).

  - There SHOULD be at least one Scrum Master for all project teams.
  - A single Development Team SHOULD NOT exceed nine members.
  - If there is only one Scrum Master, they SHOULD NOT also be a Product Owner. However, this MAY be the case where it is unavoidable.
  - If there are two or more Scrum Masters, a Scrum Master SHOULD NOT be assigned to a project where they are also Product Owner.
  - In a project team of two or more members, the Product Owner SHOULD NOT be part of the Development Team.
  - All sprints MUST begin with a planning session.
  - All sprints MUST end with a sprint retrospective.
  - All sprints SHOULD end with a sprint review.
  - Tasks SHOULD be formulated as user stories by the Product Owner.
- Kanban boards SHALL be used for capacity management.
  - These boards MAY be physical (e.g., whiteboards and sticky notes) or MAY be virtual (e.g., Trello).
  - Only a Product Owner SHOULD be able to add tasks to the product backlog.
  - The buckets used SHOULD be as follows:
    1. 'Ready', for completed user stories;
    2. 'Investigate', for user stories being converted into feature specifications;
    3. 'Develop', for features being developed from specifications;
    4. 'Test', for developed features being tested;
    5. 'Deploy', for features that have passed their tests and need to be deployed; and
    6. 'Delivered', for features that have been delivered to customers.
  - With the exception of the 'Ready' and 'Delivered' buckets, bucket capacity SHOULD be capped at the number of developers on the project.
  - The 'Investigate' bucket capacity SHOULD be twice the number of developers.
  - Tickets MUST be able to move both left or right on the board.
- Continuous deployment SHALL be used for all development.
  - Test-Driven Development MUST be used for all feature development.
  - If test coverage is ever less than 85 % of the codebase, the next sprint SHOULD be devoted to test-writing.
  - All developers SHOULD commit their code to `main` at least once per day.
  - Where the use of branches is unavoidable, developers MUST squash and rebase their changes before merging with `main`.
  - Where a commit causes production to fail, the developer responsible MUST immediately attempt to resolve the issue.
  - If, after one minute, the issue is not resolved, the rest of the Development Team MUST stop what they are doing and work on the issue until it is resolved.
- Telemetry SHALL be created by all features and environments.

  - If monitoring coverage is ever less than 85 % of the codebase, the next sprint MUST be devoted to adding telemetry.
  - Production environments MUST produce telemetry.
  - Development and staging environments SHOULD produce telemetry.

- A just culture MUST be created and maintained.
  - A blameless post-mortem MAY be scheduled after any incident.
  - A blameless post-mortem MUST be scheduled following an incident that results in any of the following:
    * production experiencing more than five minutes of downtime;
    * a MUST rule stated here being broken;
    * complaints from more than one customer;
    * an inability to deliver new code to customers;
    * a financial loss to the company; or
    * damage to the company's reputation.
  - A blameless post-mortem SHOULD be scheduled within one week of the incident being resolved.
  - A blameless post-mortem MUST NOT take place more than three weeks after the incident is resolved.
  - A blameless post-mortem SHOULD include:
    * the people involved in decisions that may have contributed to the problem;
    * the people who identified the problem;
    * the people who responded to the problem;
    * the people who diagnosed the problem;
    * the people who were affected by the problem; and
    * anyone else who is interested in attending.
  - Blameless post-mortem reports MUST be accessible by other employees and any customers affected by the incident.
  - Blameless post-mortem reports SHOULD be made public.
  - Production failures SHOULD be injected unpredictably, to ensure the resilience of the system.
  - All projects SHOULD schedule a Game Day at least once every six months.

- These are default settings. All project teams SHOULD review and MAY amend these as they see fit, but MUST explicitly record justification for their changes and, where possible, back their decisions up with data.

As said, this is the methodology that shall be used for the development of this KTP's threat intelligence solution. Pre-existing projects (i.e., brown field projects) may find some elements prohibitively difficult to implement. These teams are free to hew as close or as far to this structure as they like, but are nonetheless advised to begin planning their transitions towards Agile development practices as soon as possible.

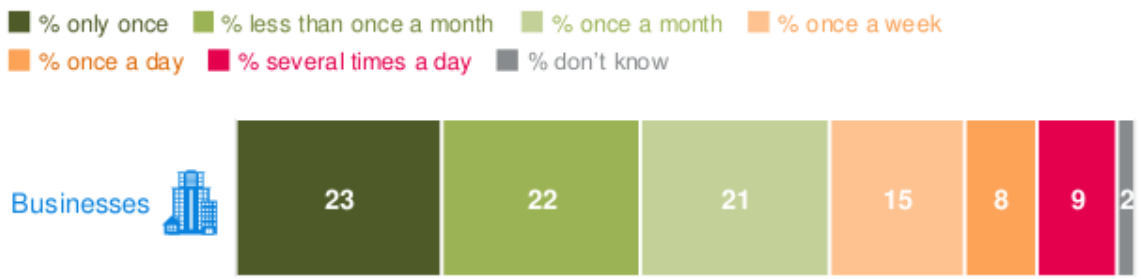


Figure 29: How often organisations have experiences breaches or attacks experiences in the last 12 months.[184]

3 ISTI Design

This section summarises some of the design and implementation decisions that went into the development of this project's prototype TIMS, prior to its unexpected cancellation.

3.1 Theory

Theoretically, this sprint formalised the mathematical underpinning for the planned risk calculation feature.[181] This mathematical underpinning shall now be described in detail: first, how to calculate the likelihood of an attack/breach; second, how to calculate the cost; and third, how to apply both of those to a given business to provide an actionable calculation of risk. The methodology is not perfect, but it suffices for the purposes of proof-of-concept.

For demonstration purposes, we decided to use the 2020 edition of the UK government's *Cyber Security Breaches Survey* (CSBS) as our first source of TI data.[182]

3.1.1 Likelihood

The CSBS does not provide data individual responses; only aggregate data. For example, fig. 29 shows the answers to 'How often organisations have experiences breaches or attacks experiences in the last 12 months', categorising the 748 respondents' answers as: 'only once'; 'less than once a month'; 'once a month'; 'once a week'; 'once a day'; 'several times a day'; and 'don't know'.

This presents an obstacle to the goal of creating a log-normal distribution of breach frequency. However, we can approximate this distribution, as breach statistics follow a Pareto distribution. First, we define the aforementioned categories numerically (adding one for 'none' and dropping the 'don't know' response):

[181] This was aided by a statistician colleague of DP's (Chris Sherlock, Dan data (28 October 2020)).

[182] Department of Digital, Media, Culture & Sport and Matt Warman, 'Cyber Security Breaches Survey 2020' (25 March 2020) ⟨https://www.gov.uk/government/statistics/cyber-security-breaches-survey-2020⟩ accessed 13 November 2020.

$$\text{Categories}_{\text{Textual}} = (\text{None}, \text{annually}, < \text{monthly}, \text{monthly}, \text{weekly}, \text{daily}, > \text{daily}) \quad (1)$$

$$\text{Categories}_{\text{Numeric}} = (0, 1, 2\text{–}11, 12, 52, 365, > 365) \quad (2)$$

Then, we can make some reasonable assumptions about the actual numbers covered by some of the categories. For example, we can assign 2–8 as the range covered by 'less than once a month', assuming that more than 8 breaches in a year would likely be remembered as 'monthly'. Similarly, we can assign the range 9–18 to 'once a month' as respondents with 13–18 breaches might round down from 'once a week'.

This gives us the following boundaries, choosing an arbitrary maximum of 8,000 (i.e., 21/day):

$$\text{Boundaries} = (1, 2, 8, 18, 80, 400, 8000) \quad (3)$$

We can then insert the values from the CSBS (where 54 % of respondents reported no breaches)[185], giving us

$$\text{Probabilities} = (0.1058, 0.1012, 0.0966, 0.069, 0.0368, 0.0414) \quad (4)$$

The cumulative distribution function of a Pareto distribution is

$$F(x) = \begin{cases} 1 - \left(\frac{b}{x}\right)^{\alpha} & x \geq b, \\ 0 & x < b. \end{cases} \quad (5)$$

and therefore, for $x \geq b$

$$1 - F(x) = \left(\frac{b}{x}\right)^{\alpha} \quad (6)$$

$$\log(1 - F(x)) = \alpha \log b - \alpha \log x \quad (7)$$

Plotting this (see fig. 30) shows us that we have a (roughly) straight line to fit.

Having plotted a linear model based on this, we can then run a number of simulations using the below formula (where u is a uniform random number)

[185] Whilst 1,348 UK organisations responded to the CSBS, and 620 (46 %) reported having identified breaches or attacks in the last 12 months ((Department of Digital, Media, Culture & Sport and Warman [n **??**] fig. 5.1)), the frequency question is apparently based on the responses of 748 organisations (i.e., 121 % of those reporting breaches). We have chosen to apply the percentages from the frequency question to the numbers from the overall question (i.e., 23 % of 46 % of businesses—12.43 %–reported one breach in the last 12 months).

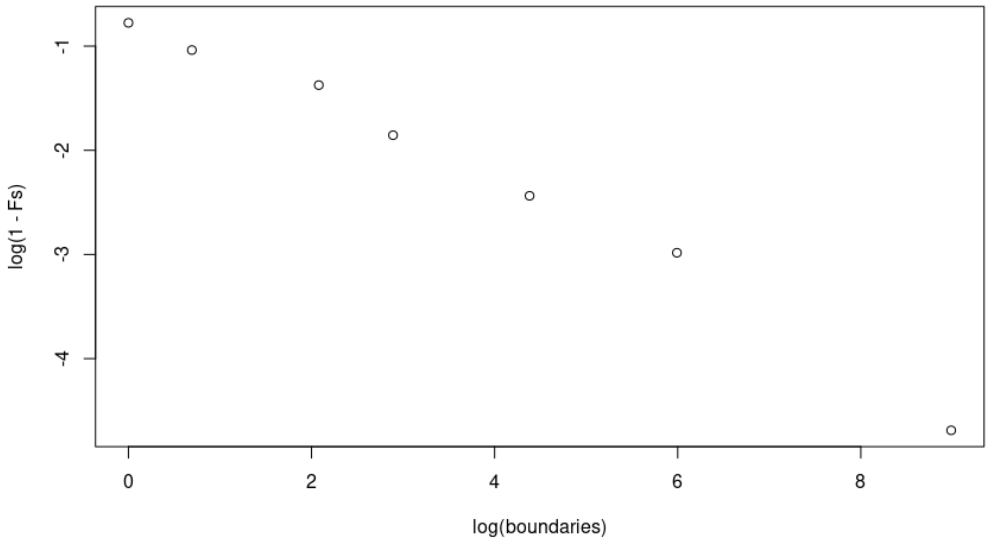


Figure 30: Testing for a line to fit.

$$x = \frac{b}{(1-u)^{\frac{1}{\alpha}}} \tag{8}$$

On an example run of 10,000 iterations, this produced the below probability distribution (with the CSBS distribution and difference for comparison)

$$\text{Probabilities}_{\text{Calculated}} = (0.1067, 0.1457, 0.0627, 0.0718, 0.0410, 0.0479) \tag{9}$$

$$\text{Probabilities}_{\text{CSBS}} = (0.1058, 0.1012, 0.0966, 0.069, 0.0368, 0.0414) \tag{10}$$

$$\text{Difference} = (-0.0158, 0.0009, 0.0445, -0.0339, 0.0028, 0.0042, 0.0065) \tag{11}$$

$$\mu\,(\text{Difference}) = 0.001 \tag{12}$$

Rounding the random numbers down, we can then plot a histogram of the number of attacks (see fig. 31).

3.1.2 Costs

In terms of calculating the cost of each breach, the CSBS provides the average cost of all breaches identified in the last 12 months (and the average cost of only those breaches with an outcome),[186] and then delves into the average direct, recovery and long-term cost figures.[187] However, these breakdown figures are only for what the respondents have identified

[186] Department of Digital, Media, Culture & Sport and Warman (n **??**) table 5.1.
[187] ibid tables 5.2–4.

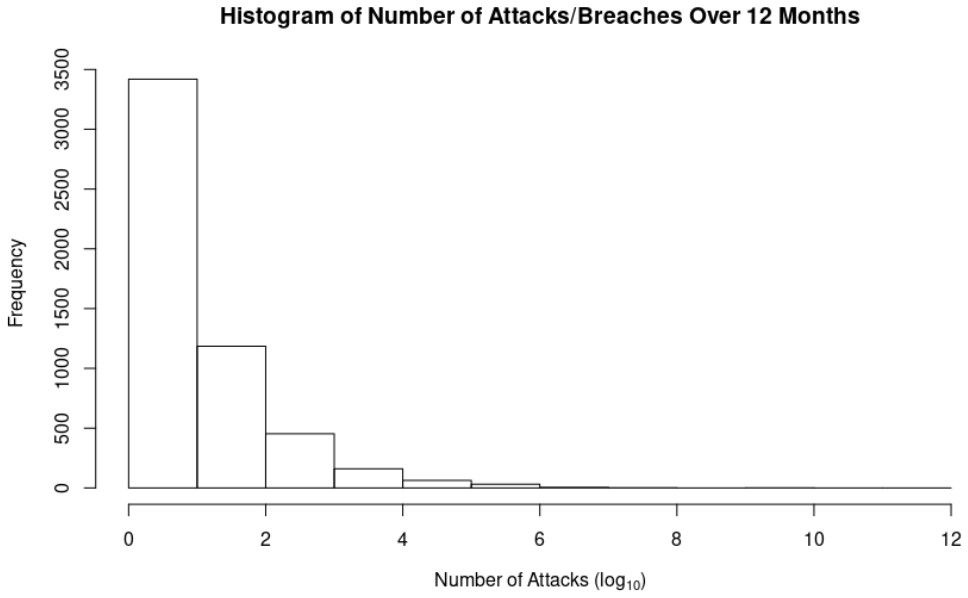


Figure 31: Generated attacks histogram.

as the 'most disruptive' breach or attack they have experienced in the last 12 months. This poses us the challenge of deriving from these figures the likely per-breach direct, recovery and long-term cost probability distributions.

For now, though, we can stick to the first set of data: the average cost of all breaches that led to an outcome.

To plot a log-normal distribution, we need to have the mean (μ) to position the centre of the curve and the standard deviation (σ) to determine its width. The CSBS provides us with the mean, and we can use these along with the median ($\mu_{\frac{1}{2}}$) to determine the $\ln(\sigma)$, and from that the σ:

$$\ln(\sigma) = \sqrt{2 \times \left(\ln(\mu) - \ln(\mu_{\frac{1}{2}})\right)} \tag{13}$$

$$\sigma = e^{\ln(\sigma)} \tag{14}$$

So, with the mean cost of £3,230 and the median of £274 from the CSBS, we can calculate that

$$\mu = 3230 \tag{15}$$

$$\sigma = 9.219401 \tag{16}$$

$$\ln(\mu) = 8.080237 \tag{17}$$

$$\ln(\sigma) = 2.22131 \tag{18}$$

From this, we can generate a probability density distribution (see fig. 32).

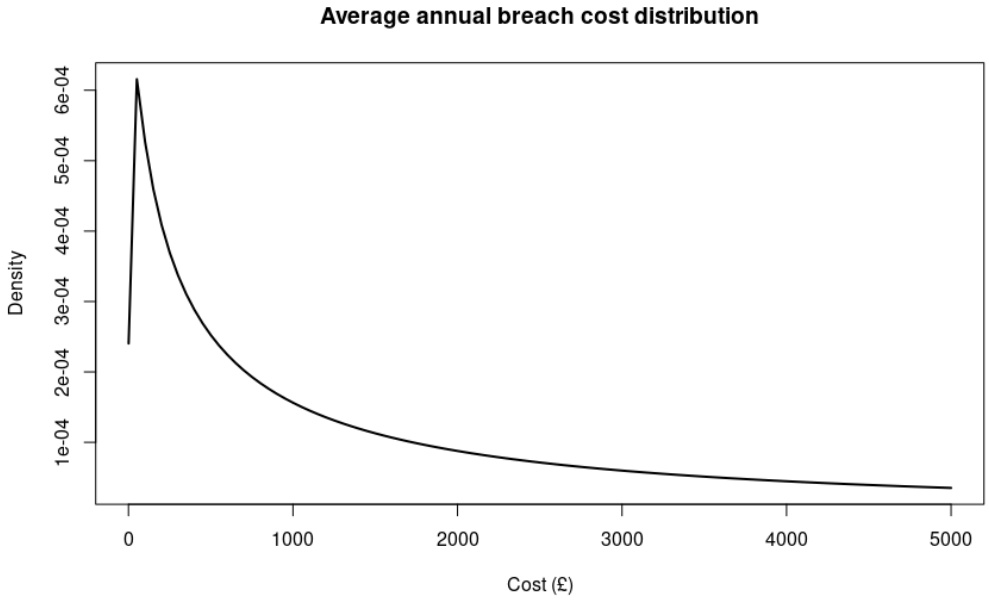


Figure 32: Generated average cost distribution.

3.1.3 Monte Carlo Simulation

Finally, we can use our likelihood and cost values to run a Monte Carlo simulation. This involves simulating n organisations over 12 months, choosing a random number of attacks for each run based on the likelihood distribution. For each attack, we than calculate a cost from the cost distribution, summing them to give us the total cost to the organisation for that year.

$$\text{Let } l = \text{number of attacks distribution} \tag{19}$$

$$\text{Let } c = \text{cost distribution} \tag{20}$$

$$L \sim l \tag{21}$$

$$C_i \sim c \tag{22}$$

$$\text{Annual Cost} = \sum_{i=0}^{L} C_i \tag{23}$$

We can plot the density of the results of this simulation as a line (see fig. 34) or a histogram (see fig. 33).

We can compare the results of this simulation to the actual average cost figures given in the CSBS for all breaches over the last 12 months (i.e., including ones without an outcome):

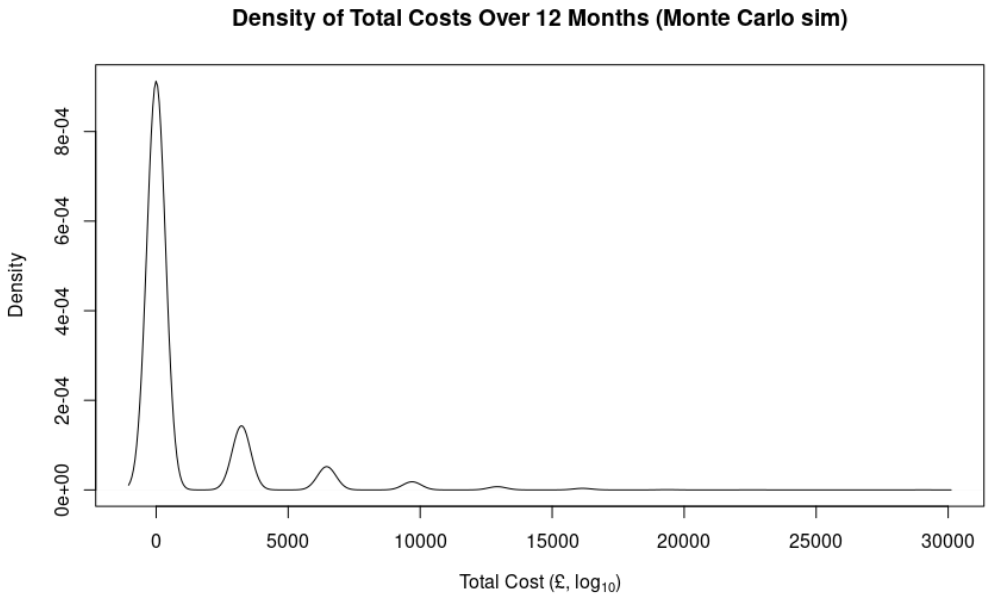


Figure 33: Monte Carlo simulation results density.

$$\mu_{\text{Monte Carlo}} = £1004.58 \tag{24}$$

$$\mu_{\frac{1}{2}\text{Monte Carlo}} = £0 \tag{25}$$

$$\mu_{\text{CSBS}} = £1010 \tag{26}$$

$$\mu_{\frac{1}{2}\text{CSBS}} = £0 \tag{27}$$

3.1.4 Loss Exceedance

Finally, we can use these simulation results to produce a loss exceedance curve (LEC) for the organisation. This allows them to specify a cost amount and calculate what the chances of them them hitting or exceeding it over the next 12 months. For example, fig. 35 shows such a curve, given a maximum cost value of £2,500.

3.2 Pre-Cancellation Plan

After a lengthy suspension due to the COVID-19 pandemic, we devised a new set of deliverables that would take us through to the end of the project and deliver a strong basis for further development. Unfortunately, the project was cancelled shortly afterwards, so we were never afforded the opportunity to fulfil these ambitions. However, they detail the methodology we had intended to use and may be of some use in the development of future systems.

The project milestones were identified, along with their achievement dates, as follows:

1. By the end of Dec 2020 we should have: a system that can run generic Monte Carlo simulations and determine baseline LECs for organisations; recalculating them both

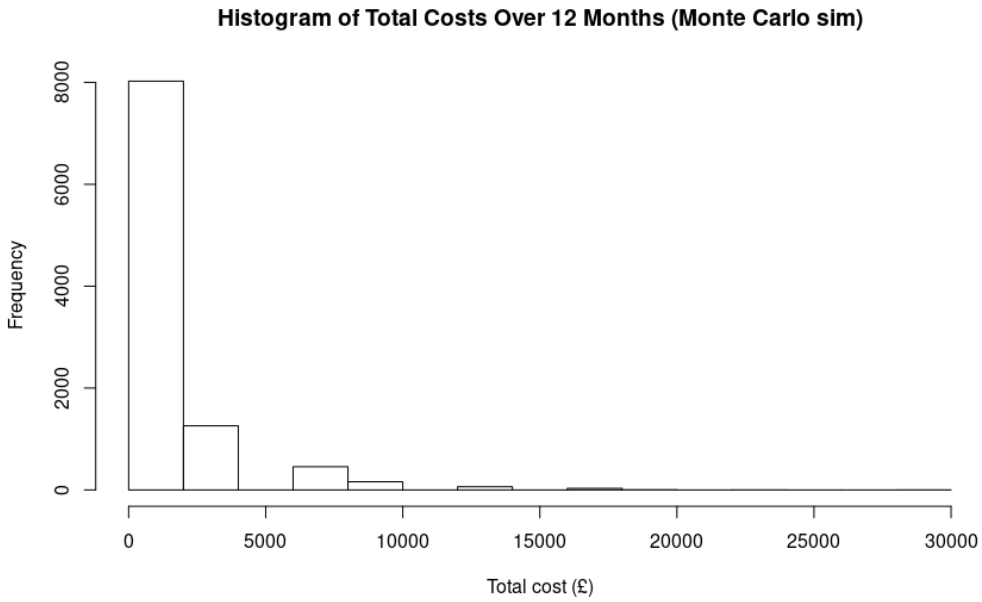


Figure 34: Monte Carlo simulation results histogram.

on-demand and at scheduled points to ensure that they always reflect the latest and most-specific information available; and implement a method and workflow by which employees can record new items of ISTI to be included in those calculations, subject to approval.

2. By the end of Mar 2021, we should have: initial, empirical evidence for the effectiveness of three controls tied to other products and services (e.g., e-learning, penetration tests, Cyber Essentials (CE) certification); conducted expert surveys and identified additional data sources to supplement the CSBS; conducted controlled A/B trials using real users.

3. By the end of May 2021, we should have: much more details simulation results, which take into account the impacts of an organisation's implemented controls (derived from an asset log), their characteristics (e.g., size, industry, etc.) and the likelihood of specific types of breach.

4. By the end of Jul 2021, we should have: the ability to perform rudimentary 'what if?' analysis, provided predictive return on investment (RoI) values for new controls and recommending the most effective controls to consider next.

5. By the end of Sep 2021 (i.e., the end of the project), we should have: the produced system documented and handed over for future development.

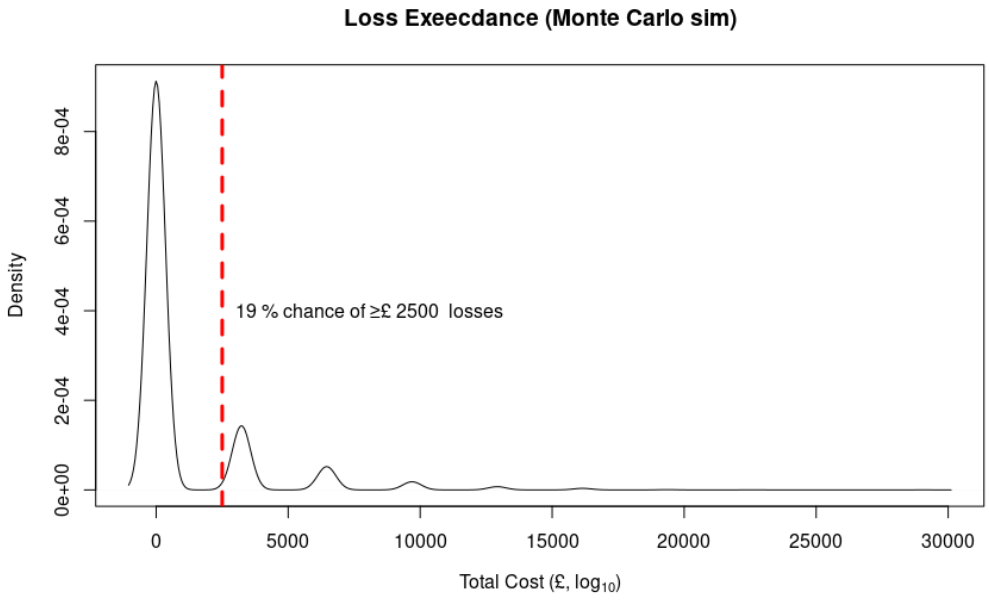


Figure 35: Monte Carlo simulation results loss exceedance.

A Relevant ISO/IEC 27000 Controls

This appendix features all of the ISO/IEC 27000-series controls that we consider to be relevant to the design of our TIMS. Those that would have to be implemented at company-level (e.g., much of clause 5) have been elided, although these may come with obligations for our project (e.g., ISO/IEC 27010 augments control 5.1.1 to highlight the need for '[a]n information sharing policy [that] should define how the community members will work together to set security management policies and direction for the information sharing community'[188]).

Unless otherwise specified, all controls below are from Annex A of the ISO/IEC 27001:2013 standard.[189]

5 Information security policies
5.1 Management direction for information security
Objective: To provide management direction and support for information security in accordance with business requirements and relevant laws and regulations.

[188] ISO/IEC 27010:2015 (n 54) p. 4–5.
[189] ISO/IEC 27001:2013 (n 37).

5.1.1	Policies for information security	*Control* An information sharing policy should define how the community members will work together to set security management policies and direction for the information sharing community. It should be made available to all employees involved in information sharing within the community. The policy may restrict its dissemination to other employees of community members.[190] The information sharing policy should define the information marking and distribution rules used within the community.[191]
7 Human resource security		
7.2 During employment		
Objective: To ensure that employees and contractors are aware of and fulfil their information security responsibilities.		
7.2.2	Information security awareness, education and training	*Control* All employees of the organization and, where relevant, contractors shall receive appropriate awareness education and training and regular updates in organizational policies and procedures, as relevant for their job function.
8 Asset management		
8.1 Responsibility for assets		
Objective: To identify organizational assets and define appropriate protection responsibilities.		

[190] ISO/IEC 27010:2015 (n 54) p. 4–5.
[191] ibid p. 4–5.

8.1.1	Inventory of assets	*Control* Information, other assets associated with information and information processing facilities shall be identified and an inventory of these assets shall be drawn up and maintained. The cloud service customer's inventory of assets should account for information and associated assets stored in the cloud computing environment.[192] The inventory of assets of the cloud service provider should explicitly identify: cloud service customer data; [and] cloud service derived data[193]
8.1.2	Ownership of assets	*Control* Assets maintained in the inventory shall be owned.
8.1.3	Acceptable use of assets	*Control* Rules for the acceptable use of information and of assets associated with information and information processing facilities shall be identified, documented and implemented. Information provided by other members of an information sharing community is an asset and should be protected, used and disseminated in accordance with any rules set by the information sharing community or by the originator.[194]
8.2 Information classification		
Objective: To ensure that information receives an appropriate level of protection in accordance with its importance to the organisation.		
8.2.1	Classification of information	*Control* Information shall be classified in terms of legal requirements, value, credibility, priority, criticality and sensitivity to unauthorised disclosure or modification.[195] See ISO 27799 for discussion of the flexibility needed to accommodate personal health information classification.[196]

[192] ISO/IEC 27017:2015: Information technology — Security techniques — Code of practice for information security controls based on ISO/IEC 27002 for cloud services (2015) p. 7.
[193] ibid p. 7.
[194] ISO/IEC 27010:2015 (n 54) p. 6.
[195] ISO/IEC 27010:2015 (n 54) p. 6.
[196] ISO 27799:2016: Health informatics — Information security management in health using ISO/IEC 27002 (2016) pp. 14–15.

8.2.2	Labelling of information	*Control* An appropriate set of procedures for information labelling shall be developed and implemented in accordance with the information classification scheme adopted by the organisation. The cloud service customer should label information and associated assets maintained in the cloud computing environment in accordance with the cloud service customer's adopted procedures for labelling.[197] The cloud service provider should document and disclose any service functionality it provides allowing cloud service customers to classify and label their information and associated assets.[198]
8.2.3	Handling of assets	*Control* Procedures for handling assets shall be developed and implemented in accordance with the information classification scheme adopted by the organisation.
8.4 Information exchanges protection[199]		
Objective: To ensure adequate protection of information exchanges within an information sharing community.		
8.4.1	Information dissemination	*Control* Information dissemination within the receiving member should be limited, based on pre-defined dissemination markings defined by the community.
8.4.2	Information disclaimers	*Control* Each information exchange should begin with a disclaimer, listing any special requirements to follow by the recipients in addition to the normal information markings.
8.4.3	Information credibility	*Control* Each information exchange should indicate the originator's degree of confidence in the transmitted information's credibility and accuracy.

[197] ISO/IEC 27017:2015 (n 192) p. 8.
[198] ibid p. 7.
[199] ISO/IEC 27010:2015 (n 54) p. 7.

8.4.4	Information sensitivity reduction	*Control* The originator of an information exchange should indicate if the sensitivity of the information supplied will reduce after some external event, or the passage of time.
8.4.5	Anonymous source protection	*Control* A community member should remove any source identification information in any communication it originates or receives where anonymity is requested.
8.4.6	Anonymous recipient protection	*Control* With the approval of the originator, members of a community should be able to receive communications without revealing their own identities.
8.4.7	Onwards release authority	*Control* Unless it is marked for wider release, information should not be distributed beyond the information sharing community without formal approval from the originator.
9 Access control		
9.1 Business requirements of access control		
Objective: To limit access to information and information processing facilities.		
9.1.1	Access control policy	*Control* An access control policy shall be established, documented and reviewed based on business and information security requirements.
9.1.2	Access to networks and network services	*Control* Users shall only be provided with access to the network and network services that they have been specifically authorized to use.
9.2 User access management		
Objective: To ensure authorized user access and to prevent unauthorized access to systems and services.		
9.2.1	User registration and de-registration	*Control* A formal user registration and de-registration process shall be implemented to enable assignment of access rights.

9.2.2	User access provisioning	*Control* A formal user access provisioning process shall be implemented to assign or revoke access rights for all user types to all systems and services. The cloud service provider should provide functions for managing the access rights of the cloud service customer's cloud service users, and specifications for the use of these functions.[200]
9.2.3	Management of privileged access rights	*Control* The allocation and use of privileged access rights shall be restricted and controlled.
9.2.4	Management of secret authentication information of users	*Control* The allocation of secret authentication information shall be controlled through a formal management process. ...it should be noted that time pressures found in health delivery situations can make effective use of passwords difficult to employ. Many health organizations have considered the adoption of alternative authentication technologies to address this problem.[201]
9.2.5	Review of user access rights	*Control* Asset owners shall review users' access rights at regular intervals.
9.2.6	Removal or adjustment of access rights	*Control* The access rights of all employees and external party users to information and information processing facilities shall be removed upon termination of their employment, contract or agreement, or adjusted upon change.
9.3 User responsibilities		
Objective: To make users accountable for safeguarding their authentication information.		
9.3.1	Use of secret authentication information	*Control* Users shall be required to follow the organization's practices in the use of secret authentication information.
9.4 System and application access control		
Objective: To prevent unauthorized access to systems and applications.		

[200] ISO/IEC 27017:2015 (n 192) p. 9.
[201] ISO 27799:2016 (n 196) p. 20.

9.4.1	Information access restriction	*Control* Access to information and application system functions shall be restricted in accordance with the access control policy. The cloud service customer should ensure that access to information in the cloud service can be restricted in accordance with its access control policy and that such restrictions are realized.[202] The cloud service provider should provide access controls that allow the cloud service customer to restrict access to its cloud services, its cloud service functions and the cloud service customer data maintained in the service.[203]
9.4.2	Secure log-on procedures	*Control* Where required by the access control policy, access to systems and applications shall be controlled by a secure log-in procedure.
9.4.3	Password management system	*Control* Password management systems shall be interactive and shall ensure quality passwords.
9.4.4	Use of privileged utility programs	*Control* The use of utility programs that might be capable of overriding system and application controls shall be restricted and tightly controlled.
9.4.5	Access control to program source code	*Control* Access to program source code shall be restricted.
10 Cryptography		
10.1 Cryptographic controls		
Objective: To ensure proper and effective use of cryptography to protect the confidentiality, authenticity and/or integrity of information.		

[202] ISO/IEC 27017:2015 (n 192) p. 1.
[203] ibid p. 1.

10.1.1	Policy on the use of cryptographic controls	*Control* A policy on the use of cryptographic controls for protection of information shall be developed and implemented. Cryptographic techniques can also be used to implement the dissemination rules of information sharing.[204]
10.1.2	Key management	*Control* A policy on the use, protection and lifetime of cryptographic keys shall be developed and implemented through their whole lifecycle.
12 Operations security		
12.1 Operational procedures and responsibilities		
Objective: To ensure correct and secure operations of information processing facilities.		
12.1.1	Documented operating procedures	*Control* Operating procedures shall be documented and made available to all users who need them.
12.1.2	Change management	*Control* Changes to the organization, business processes, information processing facilities and systems that affect information security shall be controlled. The cloud service provider should provide the cloud service customer with information regarding changes to the cloud service that could adversely affect the cloud service.[205]
12.1.3	Capacity management	*Control* The use of resources shall be monitored, tuned and projections made of future capacity requirements to ensure the required system performance. The cloud service provider should monitor the total resource capacity to prevent information security incidents caused by resource shortages.[206]

[204] ISO/IEC 27010:2015 (n 54) p. 9.
[205] ISO/IEC 27017:2015 (n 192) p. 13.
[206] ibid p. 13–14.

12.1.4	Separation of development, testing and operational environments	*Control* Development, testing, and operational environments shall be separated to reduce the risks of unauthorized access of changes to the operational environment. Where the use of PII for testing purposes cannot be avoided a risk assessment should be undertaken.[207]
12.3 Backup		
Objective: To protect against loss of data.		
12.3.1	Information backup	*Control* Backup copies of information, software and system images shall be taken and tested regularly in accordance with an agreed backup policy. Information processing systems based on the cloud computing model introduce additional or alternative mechanisms to off-site backups for protecting against loss of data, ensuring continuity of data processing operations, and providing the ability to restore data processing operations after a disruptive event.[208] PII-specific responsibilities in this respect can lie with the cloud service customer.[209]
12.4 Logging and monitoring		
Objective: To record events and generate evidence.		

[207] ISO/IEC 27018:2019: Information technology — Security techniques — Code of practice for protection of personally identifiable information (PII) in public clouds acting as PII processors (2019) p. 10.
[208] ISO/IEC 27018:2019 (n 207) p. 10-11.
[209] ibid p. 10-11.

12.4.1	Event logging	*Control* Event logs recording user activities, exceptions, faults and information security events shall be produced, kept and regularly reviewed. When required by the information sharing community, members should log the internal dissemination of shared information.[210] The cloud service provider should provide logging capabilities to the cloud service customer.[211] Where possible, event logs should record whether or not PII has been changed as a result of an event and by whom.[212]
12.4.2	Protection of log information	*Control* Logging facilities and log information shall be protected against tampering and authorized access. A procedure, preferably automatic, should be put in place to ensure that logged information is deleted within a specified and documented period.[213]
12.4.3	Administrator and operator logs	*Control* System administrator and system operator activities shall be logged and the logs protected and regularly reviewed.
12.4.4	Clock synchronisation	*Control* The clocks of all relevant information processing systems within an organization or security domain shall be synchronised to a single reference time source. The cloud service provider should provide information to the cloud service customer regarding the clock used by the cloud service provider's systems, and information about how the cloud service customer can synchronize local clocks with the cloud service clock.[214]

[210] ISO/IEC 27010:2015 (n 54) p. 10.
[211] ISO/IEC 27017:2015 (n 192) p. 15.
[212] ISO/IEC 27018:2019 (n 207) p. 11.
[213] ibid p. 11.
[214] ISO/IEC 27017:2015 (n 192) p. 16.

12.4.5	Monitoring of Cloud Services	*Control* The cloud service customer should have the capability to monitor specified aspects of the operation of the cloud services that the cloud service customer uses.[215]
12.6 Technical vulnerability management		
Objective: To prevent exploitation of technical vulnerabilities.		
12.6.1	Management of technical vulnerabilities	*Control* Information about technical vulnerabilities or information systems being used shall be obtained in a timely fashion, the organization's exposure to such vulnerabilities evaluated and appropriate measures taken to address the associated risk. The cloud service provider should make available to the cloud service customer information about the management of technical vulnerabilities that can affect the cloud services provided.[216]
12.7 Information systems audit considerations		
Objective: To minimise the impact of audit activities on operational systems.		
12.7.1	Information systems audit controls	*Control* Audit requirements and activities involving verification of operational systems shall be carefully planned and agreed to minimise disruptions to business processes.
12.7.2[217]	Community audit rights	*Control* Every information sharing community should specify the rights of members to audit the systems of other members and of any trusted service providers.
13 Communications security		
13.1 Network security management		
Objective: To ensure the protection of information in networks and its supporting information processing facilities.		
13.1.1	Network controls	*Control* Networks shall be managed and controlled to protect information in systems and applications.

[215] ISO/IEC 27017:2015 (n 192) p. 27.
[216] ibid p. 16.
[217] ISO/IEC 27010:2015 (n 54) p. 10–11.

13.1.2	Security of network services	*Control* Security mechanisms, service levels and management requirements of all network services shall be identified and included in network services agreements, whether these services are provided in-house or outsourced.
13.1.3	Segregation in networks	*Control* Groups of information services, users and information systems shall be segregated on networks. The cloud service provider should enforce segregation of network access for the following cases: segregation between tenants in a multi-tenant environment; [and] segregation between the cloud service provider's internal administration environment and the cloud service customer's cloud computing environment.[218] Where appropriate, the cloud service provider should help the cloud service customer verify the segregation implemented by the cloud service provider.[219]
13.2 Information transfer		
Objective: To maintain the security of information transferred within an organization and with any external entity.		
13.2.1	Information transfer policies and procedures	*Control* Formal transfer policies, procedures and controls shall be in place to protect the transfer of information through the use of all types of communication facilities.
13.2.2	Agreements on information transfer	*Control* Agreements shall address the secure transfer of business information between the organization and external parties. All information sharing communities should define information transfer agreements, and should only permit members to join the community if such agreements are signed and accepted.[220]

[218] ISO/IEC 27017:2015 (n 192) p. 17.
[219] ibid p. 17.
[220] ISO/IEC 27010:2015 (n 54) p. 11.

13.2.3	Electronic messaging	*Control* Information involved in electronic messaging shall be appropriately protected. All information sharing communities should define rules for the protection of information in transit, and only permit members to join the community if such rules are accepted and implemented by the prospective member. Any supporting entity should implement such rules internally.[221] Information sharing communities should consider implementing alternative mechanisms for information sharing that do not rely on electronic messaging, and enabling members to specify that specific messages are distributed by such other routes.[222]
13.2.4	Confidentiality or non-disclosure agreements	*Control* Requirements for confidentiality or non-disclosure agreements reflecting the organization's needs for the protection of information shall be identified, regularly reviewed and documented.
14 System acquisition, development and maintenance		
14.1 Security requirements of information systems		
Objective: To ensure that information security is an integral part of information systems across the entire lifecycle. This also includes the requirements for information systems which provide services over public networks.		
14.1.1	Information security requirements analysis and specification	*Control* The information security related requirements shall be included in the requirements for new information systems or enhancements to existing information systems.
14.1.2	Securing application services on public networks	*Control* Information involved in application services passing over public networks shall be protected from fraudulent activity, contract dispute and unauthorized disclosure and modification.

[221] ISO/IEC 27010:2015 (n 54) p. 11.
[222] ibid p. 11.

14.1.3	Protecting application services transactions	*Control* Information involved in application service transactions shall be protected to prevent incomplete transmission, mis-routing, unauthorized message alteration, unauthorized disclosure, unauthorized message duplication or replay.
14.2 Security in development and support processes		
Objective: To ensure that information security is designed and implemented within the development lifecycle of information systems.		
14.2.1	Secure development policy	*Control* Rules for the development of software and systems shall be established and applied to developments within the organization.
14.2.2	System change control procedures	*Control* Changes to systems within the development lifecycle shall be controlled by the use of formal change control procedures.
14.2.3	Technical review of applications after operating platform changes	*Control* When operating platforms are changed, business critical applications shall be reviewed and tested to ensure there is no adverse impact on organizational operations or security.
14.2.4	Restrictions on changes to software packages	*Control* Modifications to software packages shall be discouraged, limited to necessary changes and all changes shall be strictly controlled.
14.2.5	Secure system engineering principles	*Control* Principles for engineering secure systems shall be established, documented, maintained and applied to any information system implementation efforts.
14.2.6	Secure development environment	*Control* Organizations shall establish and appropriately protect secure development environments for system development and integration efforts that cover the entire system development lifecycle.
14.2.7	Outsourced development	*Control* The organization shall supervise and monitor the activity of outsources system development.

14.2.8	System security testing	*Control* Testing of security functionality shall be carried out during development.
14.2.9	System acceptance testing	*Control* Acceptance testing programs and related criteria shall be established for new information systems, upgrades and new versions.
14.3 Test data		
Objective: To ensure the protection of data used for testing.		
14.3.1	Protection of test data	*Control* Test data shall be selected carefully, protected and controlled.
15 Supplier relationships		
15.1 Information security in supplier relationships		
Objective: To ensure protection of the organization's assets that is accessible by suppliers.		
15.1.1	Information security policy for supplier relationships	*Control* Information security requirements for mitigating the risks associated with supplier's access to the organization's assets shall be agreed with the supplier and documented. The cloud service customer should include the cloud service provider as a type of supplier in its information security policy for supplier relationships.[223]
15.1.2	Addressing security within supplier agreements	*Control* All relevant information security requirements shall be established and agreed with each supplier that may access, process, store, communicate, or provide IT infrastructure components for, the organization's information. All community members should be made aware of the identities of all third parties involved in the provision of community services, in case they have objections to particular parties being involved in the handling of information they provide.[224]

[223] ISO/IEC 27017:2015 (n 192) p. 19.
[224] ISO/IEC 27010:2015 (n 54) p. 12.

15.1.3	Information and communication technology supply chain	*Control* Agreements with suppliers shall include requirements to address the information security risks associated with information and communications technology services and product the supply chain.
15.2 Supplier service delivery management		
Objective: To maintain an agreed level of information security and service delivery in line with supplier agreements.		
15.2.1	Monitoring and review of supplier services	*Control* Organizations shall regularly monitor, review and audit supplier service delivery.
16 Information security incident management		
16.1 Management of information security incidents and improvements		
Objective: To ensure a consistent and effective approach to the management of informations security incidents, including communication on security events and weaknesses.		
16.1.1	Responsibilities and procedures	*Control* Management responsibilities and procedures shall be established to ensure a quick, effective and orderly response to information security incidents. An information security incident should trigger a review by the public cloud PII processor, as part of its information security management process, to determine if a data breach involving PII has taken place.[225]

[225] ISO/IEC 27018:2019 (n 207) p. 13.

16.1.2	Reporting information security incidents	*Control* Information security events shall be reported through appropriate management channels as quickly as possible. Members of an information sharing community should consider whether detected events should be reported to other members of the community. The community should agree and publish guidance on the types of incident that will be of interest to other members.[226] The cloud service provider should provide mechanisms for: the cloud service customer to report an information security event to the cloud service provider; the cloud service provider to report an information security event to a cloud service customer; [and] the cloud service customer to track the status of a reported information security event.[227]
16.1.3	Reporting information security weaknesses	*Control* Employees and contractors using the organization's information systems and services shall be required to note and report any observed or suspected information security weaknesses in systems or services.
16.1.4	Assessment of an decision on information security events	*Control* Information security events shall be assessed and it shall be decided if they are to be classified as information security incidents.
16.1.5	Response to information security incidents	*Control* Information security incidents shall be responded to in accordance with the documented procedures.

[226] ISO/IEC 27010:2015 (n 54) p. 12.
[227] ISO/IEC 27017:2015 (n 192) p. 21.

16.1.6	Learning from information security incidents	*Control* Knowledge gained from analysing and resolving information security incidents shall be used to reduce the likelihood or impact of future incidents. Investigations based on information distributed by an information sharing community should be performed, to reduce the risks of similar incidents and develop a better understanding of the risks facing the community and any related significant information infrastructures.[228]
16.1.7	Collection of evidence	*Control* The organization shall define and apply procedures for the identification, collection, acquisition and preservation of information, which can serve as evidence.
16.1.8[229]	Early warning system	*Control* An early warning system should be deployed within the information sharing community to effectively communicate priority information as soon as it is available.
17 Information security aspects of business continuity management		
17.1 Information security continuity		
Objective: Information security continuity shall be embedded in the organization's business continuity management systems.		
17.1.1	Planning information security continuity	*Control* The organization shall determine its requirements for information security and the continuity of information security management in adverse situations, e.g. during a crisis or disaster.
17.1.2	Implementing information security continuity	*Control* The organization shall establish, document, implement and maintain processes, procedures and controls to ensure the required level of continuity for information security during an adverse situation.

[228] ISO/IEC 27010:2015 (n 54) p. 13.
[229] ibid p. 13.

17.1.3	Verify, review and evaluate information security continuity	*Control* The organization shall verify the established and implemented information security continuity controls at regular intervals in order to ensure that they are valid and effective during adverse situations.
17.2 Redundancies		
Objective: To ensure availability of information processing facilities.		
17.2.1	Availability of information processing facilities	*Control* Information processing facilities shall be implemented with redundancy sufficient to meet availability requirements.
18 Compliance		
18.1 Compliance with legal and contractual requirements		
Objective: To avoid breaches of legal, statutory, regulatory or contractual obligations related to information security and of any security requirements.		
18.1.1	Identification of applicable legislation and contractual requirements	*Control* All relevant legislative statutory, regulatory, contractual requirements and the organization's approach to meet these requirements shall be explicitly identified, documented and kept up to date for each information system and the organization. The information sharing community should take due account of any relevant agreements, laws and regulations relating to information sharing, such as anti-cartel legislation or regulations. This could prevent certain organizations joining the community, or place restrictions upon their representation.[230]
18.1.2	Intellectual property rights	*Control* Appropriate procedures shall be implemented to ensure compliance with legislative, regulatory and contractual requirements related to intellectual property rights and use of proprietary software products. The cloud service provider should establish a process for responding to intellectual property rights complaints.[231]

[230] ISO/IEC 27010:2015 (n 54) p. 14.
[231] ISO/IEC 27017:2015 (n 192) p. 23.

18.1.3	Protection of records	*Control* Records shall be protected from loss, destruction, falsification, unauthorized access and unauthorized release, in accordance with legislatory, regulatory, contractual and business requirements.
18.1.4	Privacy and protection of personally identifiable information	*Control* Privacy and protection of personally identifiable information shall be ensured as required in relevant legislation and regulation where applicable.
18.1.5	Regulation of cryptographic controls	*Control* Cryptographic controls shall be used in compliance with all relevant agreements, legislation and regulations.
18.1.6[232]	Liability to the information sharing community	*Control* Liability issues and remediation should be clarified, understood and approved by all members of an information sharing community, to address situations in which information is intentionally or unintentionally disclosed.
18.2 Information security reviews		
Objective: To ensure that information security is implemented and operated in accordance with the organizational policies and procedures.		
18.2.1	Independent review of information security	*Control* The organization's approach to managing information security and its implementation (i.e. control objectives, controls, policies, processes and procedures for information security) shall be reviewed independently at planned intervals or when significant changes occur.
18.2.2	Compliance with security policies and standards	*Control* Managers shall regularly review the compliance of information processing and procedures within their area of responsibility with the appropriate security policies, standards and any other security requirements.

[232] ISO/IEC 27010:2015 (n 54) p. 14–15.

18.2.3	Technical compliance review	*Control* Information systems shall be regularly reviewed for compliance with the organization's information security policies and standard.
The additional controls introduced by ISO/IEC 27018 are classified according to the eleven privacy principles of ISO/IEC 29100, rather than the eighteen clauses of ISO/IEC 27002. The relevant controls are reproduced below—all controls are from Annex A of the ISO/IEC 27018 standard. **P11 Information security**		
P11.2	Restriction on the creation of hardcopy material	*Control* The creation of hardcopy material displaying PII should be restricted.
P11.6	Encryption of PII transmitted over public data-transmission networks	*Control* PII that is transmitted over public data-transmission networks should be encrypted prior to transmission.
P11.8	Unique use of user ID	*Control* If more than one individual has access to stored PII, then they should each have a distinct user ID for identification, authentication and authorization purposes.
P11.9	Records of authorized users	*Control* An up-to-date record of the users or profiles of users who have authorized access to the information system should be maintained.
P11.10	User ID management	*Control* De-activated or expired user IDs should not be granted to other individuals.

Table 1: ISO/IEC 27000-series controls applicable to TIMS

B ISO/IEC/IEEE 12207 Processes

This appendix details all of the processes listed in the ISO/IEC/IEEE 12207:2017 standard, their stated purposes and the list of activities required to fulfil each. Within the standard, each activity is broken down into various tasks; for example, the 6.4.3 Decision Management process has the activity 'make and manage decisions', which is broken down into the following tasks:

a) Determine preferred alternative for each decision.

b) Record the resolution, decision rational, and assumptions.

c) Record, track, evaluate and report decisions.[233]

Tasks have here been elided for brevity, but are vital for successfully completing each activity.

All text below is from § 6 of the ISO/IEC/IEEE 12207:2017 standard.

No.	Process	Purpose	Activities
6.1 Agreement processes			
6.1.1	Acquisition process	To obtain a product or service in accordance with the acquirer's requirements.	Prepare for the acquisition. Advertise the acquisition and select the supplier. Establish and maintain an agreement. Monitor the agreement. Accept the product or service.
6.1.2	Supply process	To provide an acquirer with a product or service that meets agreed requirements.	Prepare for the supply. Respond to a request for supply of products or services. Establish and maintain an agreement. Execute the agreement. Deliver and support the product or service.
6.2 Organizational Project-Enabling processes			
6.2.1	Life Cycle Model Management process	To define, maintain and assure availability of policies, life cycle processes, life cycle models and procedures for use by the organisation with respect to the scope of the ISO/IEC/IEEE 12207 standard.	Establish the process. Assess the process. Improve the process.
6.2.2	Infrastructure Management process	To provide the infrastructure and services to projects to support organization and project objectives throughout the life cycle.	Establish the infrastructure. Maintain the infrastructure.

[233] ISO/IEC/IEEE 12207:2017 (n 172) p. 44.

6.2.3	Portfolio Management process	To initiate and sustain necessary, sufficient and suitable projects in order to meet the strategic objectives of the organisation.	Define and authorize projects. Evaluate the portfolio of projects. Terminate projects.
6.2.4	Human Resource Management process	To provide the organization with necessary human resources and to maintain their competencies, consistent with business needs.	Identify skills. Develop skills. Acquire and provide skills.
6.2.5	Quality Management process	To assure that products, services and implementations of the quality management process meet organisation and project quality objectives and achieve customer satisfaction.	Plan quality management. Evaluate quality management. Perform corrective and preventative action.
6.2.6	Knowledge Management process	To create the capability and assets that enable the organisation to exploit opportunities to re-apply existing knowledge.	Plan knowledge management. Share knowledge and skills throughout the organisation. Share knowledge assets throughout the organisation. Manage knowledge, skills and knowledge assets.
6.3 Technical Management processes			
6.3.1	Project Planning process	To produce and coördinate effective and workable plans.	Define the project. Plan project and technical management. Activate the project.
6.3.2	Project Assessment and Control process	To assess if the plans are aligned and feasible; determine the status of the project; and direct execution to help ensure that the performance is according to plans and schedules.	Plan for project assessment and control. Assess the project. Control the project.

6.3.3	Decision Management process	To provide a structured, analytical framework for objectively identifying, characterising and evaluating a set of alternatives for a decision.	Prepare for decisions. Analyse the decision information. Make and manage decisions.
6.3.4	Risk Management process	To identify, analyse, treat and monitor the risks continually.	Plan risk management. Manage the risk profile. Analyse risks. Treat risks. Monitor risks.
6.3.5	Configuration Management process	To manage and control system elements and configurations over the life cycle.	Plan configuration management. Perform configuration management. Perform configuration change management. Perform release control. Perform configuration status accounting. Perform configuration evaluation.
6.3.6	Information Management process	To generate, obtain, confirm, transform, retain, retrieve, disseminate and dispose of information.	Prepare for information management. Perform information management.
6.3.7	Measurement process	To collect, analyse and report objective data and information to support effective management and demonstrate the quality of the projects, services and processes.	Prepare for measurement. Perform measurement.
6.3.8	Quality Assurance process	To help ensure the effective application of the organisation's Quality Management process to the project.	Prepare for quality assurance. Perform product or service evaluations. Perform process evaluations. Manage QA records and reports. Treat incidents and problems.
6.4 Technical processes			

6.4.1	Business or Mission Analysis process	To define the business or mission problem or opportunity, characterise the solution space and determine potential solutions.	Prepare for business or mission analysis. Define the problem or opportunity space. Characterise the solution space. Evaluate alternative solution classes. Mange the business or mission analysis.
6.4.2	Stakeholder Needs and Requirements Definition process	To define the stakeholder requirements for a system that can provide the capabilities needed by users and other stakeholders in a defined environment.	Prepare for stakeholder needs and requirements definition. Define stakeholder needs. Develop the operational concept and other life cycle concepts. Transform stakeholder needs into stakeholder requirements. Analyse stakeholder requirements. Manage the stakeholder needs and requirements definition.
6.4.3	System/Software Requirements Definition process	To transform the stakeholder, user-oriented view of desired capabilities into a technical view of a solution that meets the operational needs of the user.	Prepare for system/software requirements definition. Define system/software requirements. Analyse system/software requirements. Manage system/software requirements.
6.4.4	Architecture Definition process	To generate system architecture alternatives, to select one or more and to express this in a set of consistent views.	Prepare for architecture definition. Develop architecture viewpoints. Develop models and views of candidate architectures. Relate the architecture to design. Assess architecture candidates. Manage the selected architecture.

6.4.5	Design Definition process	To provide sufficient detailed data and information about the system and its elements to enable the implementation consistent with architectural entities as defined in models and views of the system architecture.	Prepare for software system design definition. Establish designs related to each software system element. Assess alternatives for obtaining software system elements. Manage the design.
6.4.6	System Analysis process	To provide a rigorous basis of data and information for technical understanding to aid decision-making across the life cycle.	Define the system analysis strategy and prepare for system analysis. Perform system analysis. Manage the system analysis.
6.4.7	Implementation process	To realise a specified system element.	Prepare for implementation. Perform implementation. Manage results of implementation.
6.4.8	Integration process	To synthesize a set of system elements into a realised system that satisfies system/software requirements, architecture and design.	Prepare for integration. Perform integration. Manage results of integration.
6.4.9	Verification process	To provide objective evidence that a system or system element fulfils its specified requirements and characteristics.	Prepare for verification. Perform verification. Manage results of verification.
6.4.10	Transition process	To establish a capability for a system to provide services specified by stakeholder requirements in the operational environment.	Prepare for the software system transition. Perform the transition. Manage results of transition.
6.4.11	Validation process	To provide objective evidence that the system, when in use, fulfils its business or mission objectives and stakeholder requirements, achieving its intended use in its intended operational environment.	Prepare for validation. Perform validation. Manage results of validation.

6.4.12	Operation process	To use the system to deliver its services.	Prepare for operation. Perform operation. Manage results of operation. Support the customer.
6.4.13	Maintenance process	To sustain the capability of the system to provide a service.	Prepare for maintenance. Perform maintenance. Perform logistics support. Manage results of maintenance and logistics.
6.4.14	Disposal process	To end the existence of a system element or system, appropriately handle replaced or retired elements and to properly attend to identified critical disposal needs.	Prepare for disposal. Perform disposal. Finalise the disposal.

Table 2: ISO/IEC 12207 process implementation

List of Figures

List of Tables

Acronyms

API Application Programming Interface. 39

AWS Amazon Web Services. 44

CE Cyber Essentials. 56

CIA Central Intelligence Agency. 14

CNI critical national infrastructure. 5, 6

COVID-19 coronavirus disease 2019. 55

CSBS *Cyber Security Breaches Survey*. 50–56

CSIRT computer security incident response team. 4

CTI cyber threat intelligence. 27

CVE Common Vulnerabilities and Exposures. 24

CyBoK Cyber Security Body of Knowledge. 14

DLM Detection Maturity Level. 27

DPA 2018 Data Protection Act 2018. 2, 4

DSP digital service provider. 5

EO Executive Order. 20

EU European Union. 2, 4, 20

F3EAD Find, Fix, Finish, Exploit, Analyse and Disseminate. 16

GDPR General Data Protection Regulation. 2–8, 84, 88

GL grey literature. 13, 21

GUI Graphical User Interface. 39

IaC Infrastructure as Code. 39

ICO Information Commissioner's Office. 5

ID IDentifier. 77

IDS intrusion detection service. 23

IEC International Electrotechnical Commission. 1, 2, 6–12, 30, 44, 45, 47, 57, 77, 78, 83, 84, 86

IEEE Institute of Electrical and Electronics Engineers. 2, 30, 44, 45, 77, 78, 84

IoC indicator of compromise. 14, 27

IP Internet Protocol. 18

IS intelligence studies. 14

ISMS information security management system. 6, 8–11

ISO International Organization for Standardization. 1, 2, 6–12, 30, 44, 45, 47, 57, 77, 78, 83, 84, 86

ISTI information security threat intelligence. 1, 10, 12, 14, 17–30, 56, 84

IT information technology. 1, 13, 14, 39

KTP Knowledge Transfer Partnership. 1, 47, 49

LEC loss exceedance curve. 55

MLR multivocal literature review. 1, 12, 13, 21

MRTI machine-readable TI. 23

MSS managed security service. 22

MVP Minimum Viable Product. 33, 42

NCSC National Cyber Security Centre. 4

NIS Directive Network Information Security Directive. 4–6, 20

NIS Regulations Network and Information Systems Regulations 2018. 4, 6

OES operator of essential services. 5

PDCA Plan-Do-Check-Act. 10

PII personally-identifiable information. 6, 7, 10, 65, 66, 72, 77, 84

PPD Presidential Policy Directive. 20

QA Quality Assurance. 39, 80

RoI return on investment. 56

RQ research question. 13

SaaSS Service-as-a-Software-Substitute, a.k.a. Software-as-a-Service (SaaS). 1

SLR systematic literature review. 13

SOA Service-Oriented Architecture. 42

STIX™ Structured Threat Information eXpression. 18, 24

TDD Test-Driven Development. 39, 48

TI threat intelligence. 18, 21–23, 50, 84, 87

TIaaS Threat Intelligence-as-a-Service. 21

TIMS threat intelligence management system. 1, 2, 6, 11, 12, 17, 18, 23–25, 50, 57, 77, 86

TLP Traffic Light Protocol. 12

TTI technical threat intelligence. 17, 18

UK United Kingdom. 1, 3–5, 50, 51

US United States. 14, 20

XP Extreme Programming. 2, 30, 34–36

YAML YAML Ain't Markup Language. 39

Glossary

development team Within Scrum, the three to nine members of the team who decide upon technical solutions and implement them. 38

information radiator An Agile term for a highly-visible display that allow anybody walking past to view key product metrics at a glance.. 42

personal data Defined within the GDPR as 'any information relating to an identified or identifiable natural person ("data subject")'. 6

personally-identifiable information Defined by the US Government's Office of Privacy and Open Government as 'information which can be used to distinguish or trace an individual's identity...alone, or when combined with other personal or identifying information which is linked or linkable to a specific individual'. 6

product backlog Within Scrum, the collection of tasks to complete managed by the product owner. 38

product owner Within Scrum, the project member who represents the product's stakeholders and is responsible for the results. 38

Scrum An Agile methodology of managing complex knowledge work by dividing it into sprints. 37–39

Scrum Master Within Scrum, the project member responsible for facilitating the Scrum process. 38

sprint Evaluating the results of a sprint and identifying lessons for future sprints. 38

sprint Demonstrating completed work to stakeholders at the end of a sprint. 38, 39

sprint A short (one week to one month), timeboxed interval of development that is planned beforehand and reviewed afterwards. 37, 38

sprint backlog Within Scum, the collection of tasks to be completed during the current sprint. 38

technical debt The gradual accumulation of workarounds, good-enoughs and bodge jobs during software development.. 43

References

Adams RJ, Smart P, and Huff AS, 'Shades of grey: guidelines for working with the grey literature in systematic reviews for management and organizational studies' (2017) 19(4) International Journal of Management Reviews 432.

Beck K, *Extreme Programming Explained: Embrace Change* (Second, first published 1999, Addison-Wesley Professional 2004).

Beck K and others, Manifesto for Agile Software Development (2001).

Bibby A, 17 Stats About Remote Work in 2019 (2019).

Bloom N and others, 'Does working from home work? Evidence from a Chinese experiment' (2014) 130(1) The Quarterly Journal of Economics 165.

Bradner S, Key words for use in RFCs to Indicate Requirement Levels (1997).

Bromander S, Jøsang A, and Eian M, 'Semantic Cyberthreat Modelling.' (2016).

Bromiley M, 'Threat intelligence: What it is, and how to use it effectively' [2016] SANS Institute InfoSec Reading Room.

Brown R and Lee RM, 'The Evolution of Cyber Threat Intelligence (CTI): 2019 SANS CTI Survey' [2017] SANS Institute InfoSec Reading Room.

Burger EW and others, 'Taxonomy model for cyber threat intelligence information exchange technologies' (2014).

CIA, The Intelligence Cycle (2001).

Conway ME, 'How do committees invent' (1968) 14(4) Datamation 28.

Dalziel H, *How to define and build an effective cyber threat intelligence capability* (Syngress 2014).

Data Protection Act 2018 2018.

Debar H, *Security Operations & Incident Management* (, CyBoK 2019).

Dekker S, *Just Culture: Balancing safety and accountability* (CRC Press 2016).

Department of Digital, Media, Culture & Sport and Warman M, 'Cyber Security Breaches Survey 2020' (25 March 2020) ⟨https://www.gov.uk/government/statistics/cyber-security-breaches-survey-2020⟩ accessed 13 November 2020.

Directive (EU) 2016/1148 of the European Parliament and of the Council of 6 July 2016 concerning measures for a high common level of security of network and information systems across the Union 2016-07-19.

Directive on the protection of individuals with regard to the processing of personal data and on the free movement of such data 1995-11-23.

Elmellas J, 'Knowledge is power: the evolution of threat intelligence' (2016) 2016(7) Computer Fraud & Security 5.

Fachkha C and Debbabi M, 'Darknet as a source of cyber intelligence: Survey, taxonomy, and characterization' (2015) 18(2) IEEE Communications Surveys & Tutorials 1197.

Falk C, Cyber Threat Intelligence as a Service (2017).

Futter A, '"Cyber" semantics: why we should retire the latest buzzword in security studies' (2018) 3(2) Journal of Cyber Policy 201.

Garousi V, Felderer M, and Mäntylä MV, 'Guidelines for including grey literature and conducting multivocal literature reviews in software engineering' (2019) 106 Information and Software Technology 101.

Gill P, 'Theories of intelligence: Where are we, where should we go and how might we proceed?' in *Intelligence Theory: Key Questions and Deabtes* (Routledge 2008).

Gill P and Phythian M, *Intelligence in an Insecure World* (Second, Polity Press 2012).

Hulnick AS, 'What's wrong with the Intelligence Cycle' (2006) 21(6) Intelligence and national Security 959.

Irrazabal E and others, 'Applying ISO/IEC 12207: 2008 with SCRUM and Agile methods' (2011).
ISO/IEC 29100:2011: Information technology — Security techniques — Privacy framework (2011).
ISO/IEC 27001:2013: Information technology — Security techniques — Information security management systems — Requirements (2013).
ISO/IEC 27010:2015: Information technology — Security techniques — Information security management for inter-sector and inter-organizational communications (2015).
ISO/IEC 27017:2015: Information technology — Security techniques — Code of practice for information security controls based on ISO/IEC 27002 for cloud services (2015).
ISO 27799:2016: Health informatics — Information security management in health using ISO/IEC 27002 (2016).
ISO/IEC 27000:2016: Information technology — Security techniques — Information security management systems — Overview and vocabulary (2016).
ISO/IEC/IEEE 12207:2017: Systems and software engineering — Software life cycle processes (2017).
ISO/IEC 27018:2019: Information technology — Security techniques — Code of practice for protection of personally identifiable information (PII) in public clouds acting as PII processors (2019).
Jacobson D, Yuan D, and Joshi N, Scryer: Netflix's predictive auto scaling engine, 'Netflix Technology Blog' (2013).
Jobsite, The rise of women in technology (2017).
Jr WB, *Practical cyber intelligence: how action-based intelligence can be an effective response to incidents* (Packt Publishing Ltd 2018).
Kent S, 'Words of estimative probability' [1964].
Kim A and Kang MH, *Determining asset criticality for cyber defense* (techspace rep, US Naval Research Laboratory 2011).
Kim G, The Three Ways: The Principles Underpinning DevOps, 'IT Revolution' (2016).
Kim G and others, *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations* (IT Revolution 2016).
Lawson C, Benson R, and Contu R, *Market Guide for Security Threat Intelligence Products and Services* (, Gartner 2019).
Littlefield A, The Beginner's Guide to Scrum and Agile Project Management (2016).
L Marinos and M Lourenço (eds), *ENISA Threat Landscape Report 2018: 15 Top Cyberthreats and Trends* (, ENISA 2019).
Mavroeidis V and Bromander S, 'Cyber threat intelligence model: an evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence' (2017).
Menges F, Sperl C, and Pernul G, 'Unifying Cyber Threat Intelligence' (2019).
Nagappan N and others, 'Realizing quality improvement through test driven development: results and experiences of four industrial teams' (2008) 13 Empirical Software Engineering 289.
Network and Information Systems Regulations 2018, SI 2004/3166 2018.
Office of Privacy and Open Government, The, 'Safeguarding Information' ⟨http://www.osec.doc.gov/opog/privacy/PII_BII.html⟩ accessed 3 December 2019.
Owl Labs, *State of Remote Working 2019* (, Owl Labs 2019).
Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) 2016-05-04.

Ries E, *The Lean Startup: How Constant Innovation Creates Radically Successful Businesses* (Portfolio Penguin 2011).
Rockstar Spouse, Wives of Rockstar San Diego employees have collected themselves, 'Gamasutra'.
Sandberg S, 'What can "lies" tell us about life? Notes towards a framework of narrative criminology' in *Advancing Qualitative Methods in Criminology and Criminal Justice* (Routledge 2014).
Sauerwein C, Sillaber C, and Breu R, 'Shadow cyber threat intelligence and its use in information security and risk management processes' [2018] Multikonferenz Wirtschaftsinformatik (MKWI 2018).
Sauerwein C and others, 'Threat intelligence sharing platforms: An exploratory study of software vendors and research perspectives' [2017].
Schreier J, Inside Rockstar Games' Culture Of Crunch, 'Kotaku'.
Schwaber K, 'SCRUM Development Process' (Sutherland Jeff and Casanave C and others eds, Springer London 1997).
Scott A, Testing Pyramids & Ice-Cream Cones (2018).
Shackleford D, 'Who's using Cyberthreat Intelligence and how?' [2015] SANS Institute.
Sherlock C, Dan data (28 October 2020).
Shoup R, 'From the Monolith to Microservices: Lessons from Google and eBay' (2016) vol 2016.
Skopik F, Settanni G, and Fiedler R, 'A problem shared is a problem halved: A survey on the dimensions of collective cyber defense through security information sharing' (2016) 60 Computers & Security 154.
Slimmon D, Do-nothing scripting: the key to gradual automation (2019).
Stillions R, 'The DML Model' (2014) ⟨https://ryanstillions.blogspot.com/2014/04/the-dml-model_21.html⟩ accessed 25 November 2019.
Surette T, EA settles OT dispute, disgruntled "spouse" outed, 'GameSpot' (2006).
Sutherland J and Schwaber K, *The Scrum Guide™: The Definitive Guide to Scrum: The Rules of the Game* (2017).
Thuret-Benoist M, 'What is the difference between personally identifiable information (PII) and personal data?' (2019) ⟨https://techgdpr.com/blog/difference-between-pii-and-personal-data/⟩ accessed 3 December 2019.
Tounsi W and Rais H, 'A survey on technical threat intelligence in the age of sophisticated cyber attacks' (2018) 72 Computers & security 212.
Wagner TD and others, 'Cyber threat intelligence sharing: Survey and research directions' (2019) 87 Computers & Security 101589.
Wells D, The Rules of Extreme Programming (1999).
White House, The, Executive Order – Improving Critical Infrastructure Cybersecurity (2013).
Presidential Policy Directive – Critical Infrastructure Security and Resilience 2013.
Wittkopf ER, Jones CM, and Kegley Jr CW, *American Foreign Policy: Pattern and Process* (Cengage Learning 2007).
Womack JP and Jones DT, *Lean thinking: banish waste and create wealth in your corporation* (Second, first published 1996, Free Press 2003).