

Cyber-Risk Indicators for Connected Medical Devices

Simeon Tverdal¹[0000-0003-1660-4127], Gencer Erdogan¹[0000-0001-9407-5748],
Andrea Neverdal Skytterholm²[0000-0001-7507-6366], Samuel M.
Senior³[0000-0002-3428-9215], Steve Taylor³[0000-0002-9937-1762], and Laura
Carmichael³[0000-0001-9391-1310]

¹ Sustainable Communication Technologies, SINTEF Digital, Oslo, Norway

² Software Engineering, Safety and Security, SINTEF Digital, Trondheim, Norway

³ IT Innovation Centre, University of Southampton, Southampton, U.K.

Abstract. Connected Medical Devices (CMDs) play a crucial role in modern healthcare, enabling real-time monitoring, automated decision making, and remote patient care. However, increasing reliance on digital connectivity introduces significant cybersecurity risks. As CMDs are used in increasingly diverse contexts, traditional risk assessment approaches may fail to capture the nuances of each use case. The increased attack surface and additional lack of cybersecurity experience among daily users further highlight the need for accurate and comprehensible risk management information. Cyber-risk indicators are additional pieces of information that can connect the risk assessment to its dynamic context, enabling risk assessors to estimate risk values more precisely and accurately. In this article, we define and present domain-specific cyber-risk indicators to facilitate dynamic risk assessment in the CMD domain and demonstrate how they can be used in the risk assessment process. Preliminary results from four real-world industry case studies are promising and validate the feasibility of our approach.

Keywords: cyber risk · indicator · cybersecurity · privacy · risk assessment · connected medical devices · case study

1 Introduction

Connected Medical Devices (CMDs) have become indispensable in contemporary healthcare, revolutionizing the way medical professionals monitor patients, make decisions, and provide care remotely. These devices facilitate the collection and analysis of real-time data, allowing timely interventions and improved patient outcomes. Additionally, CMDs support automated decision-making processes, which can enhance the accuracy and efficiency of medical treatments.

However, the growing dependence on digital connectivity in CMDs also presents substantial cybersecurity challenges [3]. As these devices become more interconnected, they become potential targets for cyberattacks, which can compromise

patient data, disrupt medical services, and even endanger patient lives. Therefore, it is imperative to carefully balance the benefits of CMDs with the associated cybersecurity risks.

As Connected Medical Devices are increasingly utilized in diverse settings, including home patient care, traditional approaches to risk assessment often fall short in addressing the unique context of each use case. The expanded attack surface, coupled with the general lack of cybersecurity expertise among everyday users, underscores the critical need for precise, current and easily understandable risk management information.

In this paper, we further explore the concept of cyber-risk indicators by utilizing four industry case studies all centred on CMDs. An indicator is defined as *a piece of information relevant for assessing risk levels*. They can help facilitate a way to bridge the gap between a risk model and the system under evaluation, providing the cyber-risk analyst with a more dynamic risk assessment process. Indicators can further inform the context of the system under evaluation by providing a more dynamic window into the realities of the system, allowing the risk model and accompanying risk evaluation to reflect the current reality with more accuracy. Ideally, these indicators should be sourced from a diverse range of inputs, including documentation and real-time monitoring tools.

This article first outlines the background of the indicator identification process (Section 2), before providing related work (Section 3), followed by detailing the methodology and presenting several example indicators (Section 4). In addition, we will discuss how to translate the source data into indicator values, offering an initial overview of the methodology. We will then introduce four industry case studies to demonstrate the results of the indicator identification process (Section 5). Finally, we conclude the paper including future work and threats to validity (Section 6).

2 Background

2.1 The CORAS method

CORAS is a risk management approach that consists of a risk assessment method in line with ISO27005, a modelling language and a tool. It falls under the umbrella of defensive risk management, meaning it concerns itself with preventing risk in contrast to weighing it against possible benefit. CORAS is developed for the express purpose of being understood by laypeople without sacrificing complexity, and includes both models showing complete attack paths, as well as simplified representations of such for the purpose of stakeholder presentation and information dissemination. In line with this concept, the risk assessment is performed by expert risk assessors in concert with stakeholders. This is done through collaborative meetings or brainstorming sessions, allowing the stakeholders to provide input and otherwise contribute to the modelling process during the entire process.

The CORAS method uses five types of models, representing different stages in the assessment process. These are *Asset models*, *Threat models*, *Risk models*,

Treatment models and *Treatment overview models*. Asset models present the assets included in the scope of the risk assessment, Threat models contain the full threat paths from threat source to asset, and include vulnerabilities and indicators. This is where most of the modelling is performed. A CORAS Risk model is an overview of the Threat model, with most detail omitted for the purpose of readability. A Treatment model is a threat model with appended treatments or mitigations, while a Treatment overview model is an overview of risks and treatment with most detail omitted. In this paper we use the more generic term *risk model* when referring to risk models in general, but the term *threat model* when referring to CORAS threat models specifically.

A typical full-scale CORAS risk assessment is carried out in eight steps [8]:

1. **Preparation for the analysis:** Define the scope, objectives, and context of the analysis, gather preliminary information, and set up the analysis team.
2. **Customer presentation of target:** Engage with stakeholders to align on the objectives, scope, and context of the risk analysis, ensuring mutual understanding and agreement.
3. **Refining the target description:** Collect detailed information about the system, context, and assets, and identify the main security objectives.
4. **Approval of target description:** Document the system and its environment thoroughly, and have stakeholders review and validate this documentation.
5. **Risk identification using threat diagrams:** Conduct brainstorming sessions and workshops with stakeholders to identify potential threats, vulnerabilities, and unwanted incidents.
6. **Risk estimation using threat diagrams:** Assess the likelihood and impact of identified risks using qualitative or quantitative methods.
7. **Risk evaluation using risk diagrams or risk evaluation matrices:** Compare estimated risks against predefined risk criteria to determine their acceptability.
8. **Risk treatment using treatment diagrams:** Develop and evaluate options for mitigating, transferring, avoiding, or accepting risks, and plan for implementing the chosen risk treatments.

The terminology used in CORAS and in this paper is explained below [8].

1. A **threat source** is a potential cause of an unwanted incident. We distinguish between deliberate human threat, accidental human threat, and non-human threat such as malware or failure.
2. A **threat scenario** is a chain or series of events that is initiated by a threat and that may lead to an unwanted incident.
3. A **vulnerability** is a weakness, flaw or deficiency that opens for, or may be exploited by, a threat to cause harm to or reduce the value of an asset.
4. An **unwanted incident** is an event that harms or reduces the value of an asset.
5. An **asset** is something to which a party assigns value and hence for which the party requires protection.

6. A **party** is an organisation, company, person, group or other body on whose behalf the risk assessment is conducted.
7. An **indicator** is a piece of information that is relevant for assessing the risk level. An indicator may be assigned to any element in a CORAS threat model.

These elements are presented in the simple model shown in Figure 1, with the numbering matching the above list. The party element is only present in asset models, and so is not displayed in this example threat model.

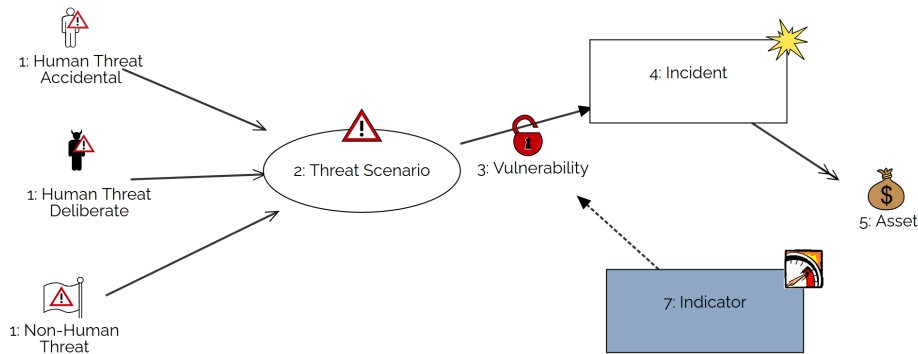


Fig. 1: CORAS threat diagram elements

2.2 Indicators

From the risk management perspective, indicators can be defined as any piece of information that can be used to assess risk levels. This encompasses information gathered during context establishment concerning the overall security of the system, as well as signals of events or situations relevant to the cybersecurity of the operation of CMDs in the context they are deployed. When discussing indicators in this paper, we are referring the *model element* representing this information, as described in Table 1. The indicator value and indicator source are referred to explicitly where relevant. Indicator source refers to where and how the indicator is obtained, while the indicator value is a metric received from the source, either directly or after processing, which is used to modify the risk value of the model.

We distinguish between four types of indicators: expert knowledge, test results, network-layer monitoring, and application-layer monitoring. Tables 2 through 5 explain these categories in more detail.

These indicator types represent potential sources of information that can be used to feed the risk assessment script with information about the likelihood or consequence of elements leading up to the unwanted incident of a risk. Moreover,

Table 1: Indicator Fields and Descriptions

Field	Description
ID	Uniquely identifies each indicator. Naming convention: IN-<Number>, where IN stands for Indicator.
Question	Defines the indicator in terms of a question that needs to be answered to obtain a value for risk assessment.
Data type	Specifies the output type of the indicator, e.g., binary (yes/no) or numerical value.
Motivation	Describes why the indicator is useful.
Indicator type	Shows the type of the indicator. The author marks the type with an “X”.
Means of obtaining indicator	Describes how the indicator may be obtained from a technical perspective.

Table 2: Expert knowledge

Indicator	Expert knowledge
Description	Indicators derived from the insights, experience, and specialized knowledge of domain experts, typically used to identify risks or vulnerabilities that may not be immediately apparent through automated tools or predefined methods.
Purpose	To incorporate human expertise and contextual understanding into risk assessment, especially for complex or novel scenarios that cannot be fully captured by standard tests or monitoring.
Examples	<ul style="list-style-type: none"> – An expert’s identification of emerging cybersecurity threats based on trends in the medical device industry. – Recommendations from a security specialist on potential vulnerabilities in custom-built systems. – Insights from clinical professionals about how device malfunctions might lead to patient harm.

the implementation of indicators in the assessed system will help facilitate a more dynamic assessment of risks.

3 Related work

This article builds on the foundational work done by Erdogan et al. (2024) [4], where a more in-depth explanation of the CORAS methodology in the context of CMDs is presented. The concept of indicators, as used in this paper, was also introduced in that earlier work.

Indicators as a concept have been discussed in the literature from various perspectives. One of the contributions most relevant to this paper is the article by Meland et al. (2021) [9], which presents a systematic literature review of

Table 3: Test result

Indicator	Test result
Description	Indicators derived from the outcomes of testing procedures, such as security tests or functional tests, to assess the state of the system or identify specific weaknesses or vulnerabilities.
Purpose	They provide information about potential vulnerabilities or failures identified through testing.
Examples	<ul style="list-style-type: none"> – Calibration tests for In Vitro Medical Device (IVD) glucose sensors reveal inaccuracies, risking improper diagnoses or treatments. – Latency tests for IVD devices highlight delays, risking data loss and delayed decisions. – Firmware integrity checks detect tampering, risking device functionality and security.

Table 4: Network-layer monitoring

Indicator	Network-layer monitoring
Description	Indicators focused on network-level activities and configurations, providing insights into network-related risks and vulnerabilities.
Purpose	Monitor and detect risks associated with network communications, such as unauthorized access, data interception, or weak encryption.
Examples	<ul style="list-style-type: none"> – Abnormal traffic patterns indicating potential Distributed Denial of Service (DDoS) attacks. – Unencrypted data transmissions between medical devices and servers. – Weaknesses in network segmentation or firewall configurations.

indicator data in the cybersecurity domain. Meland et al. found that the literature predominantly focuses on technically sourced indicators, particularly those related to network and system resources. This category of indicators has seen the most significant increase in attention and utilization.

In addition to technically sourced indicators, Meland et al. highlighted a gap in the use of domain-specific data and expert opinions. Despite their potential value, these sources of indicator data are less frequently utilized in the literature. This finding underscores the need for a more balanced approach that incorporates both technical and domain-specific indicators, as well as expert insights, to provide a more holistic view of the risk landscape.

Table 5: Application-layer monitoring

Indicator	Application-layer monitoring
Description	Indicators related to the security and functionality of applications interacting with CMDs.
Purpose	Identify vulnerabilities or threats specific to software applications and their interfaces.
Examples	<ul style="list-style-type: none"> – Use of outdated libraries or dependencies with known vulnerabilities. – Unvalidated input fields leading to potential injection attacks. – Lack of secure APIs exposing sensitive patient data.

Erdogan et al. propose a method for quantitative cyber-risk assessment using CORAS models and Bayesian Networks [5]. Indicators, derived from expert input and monitoring data, are used to dynamically adjust likelihood and impact estimates, enabling continuous and adaptive risk evaluation. The authors also present a method for developing qualitative cyber-risk assessment algorithms, combining CORAS with DEXi, a multi-criteria decision-making tool [6]. The method helps produce executable algorithms that assign qualitative risk levels based on the dynamic state of systems. It is targeted at practitioners who lack advanced programming skills but need transparent and easy-to-maintain tools. Our work is closely related to the work presented in these papers, but the usage domains are different. Our work is closely related to these contributions, but targets a different domain. While Erdogan et al. [5, 6] provide generic methods that enable dynamic risk assessment through the use of indicators and threat patterns, our focus is on systematically identifying threat scenarios and domain-specific cyber-risk indicators tailored to connected medical devices (CMDs).

Asiri et al. [2] provide a comprehensive survey of Indicators of Compromise (IOCs) in Industrial Control Systems (ICS), focusing on their role in post-incident cyber forensics and threat detection. The authors highlight that while traditional IT networks have well-established IOC practices, ICS environments pose unique challenges due to differences in architecture, limited resources, and safety-critical operations. Moreover, they analyse standards for representing IOCs, existing tools and techniques, and challenges in ICS environments, and propose a list of potential IOCs tailored to ICS attacks. The paper also maps these IOCs to common attack types and discusses how they can support incident detection and response. The authors classify IOCs into four main types: 1) Atomic Indicators (basic data points like IP addresses, URLs, filenames, etc.), 2) Computed Indicators (data derived through computation, e.g., file hashes, certificate hashes), 3) Behavioral Indicators (observable patterns of adversary behaviour (e.g., spear phishing attempts, repeated login failures), 4) Physical Measurement Indicators specific to ICS (anomalies in physical parameters, such as unexpected voltage

levels and actuator states). Our approach and the one proposed by the authors are complementary: while they leverage indicators primarily to detect and map observed information to known attack types, we use the information obtained through indicators to assess the risk level of potential attack paths.

Adaros-Boye et al. [1] introduce an Indicator of Risk (IoR) Library aimed at enabling continuous cyber-risk monitoring in Industrial Control Systems (ICS). These indicators are similar to the indicators we refer to as network-layer monitoring and application-layer monitoring. The authors seek to identify what information is necessary to monitor cyber risks and how this information can be derived from observable, measurable variables. To achieve this, they developed a library of 95 IoRs mapped to adversary techniques from the MITRE ATT&CK [10] for ICS framework.

Silvestri et al. [13] propose a method for identifying and prioritising cyber threats in healthcare ecosystems by using Natural Language Processing to extract threat information related to specific assets from widely available security-related information on the Internet, and assess risk levels based on frequency of occurrence, supporting informed mitigation actions. Our approach and theirs are complementary, as the output of their method can serve as input to the indicators in our approach, enabling dynamic risk assessment.

4 Methodology

This section explains the general methodology for identifying indicators, possible sources of indicator data, and how to translate these data into indicators of various types. Finally, the method used with the case studies will be presented.

4.1 Indicator identification

Indicator identification is done as part of an extended CORAS risk assessment. First, we describe the target of analysis and its assets. As part of this step, we also describe high-level potential threat sources, threat scenarios, vulnerabilities, and unwanted incidents. The output of this step is a high-level risk table that captures the aforementioned information.

We base ourselves on the descriptions provided by the case study partners to identify the assets that need protection and to identify the main components in the target system which need to be considered during a risk assessment. In addition, we hold regular meetings with the case study partners to align the objectives, scope, and context of the risk analysis, ensuring mutual understanding and agreement.

Next, based on high-level risk tables, we create CORAS threat models that include threat sources, threat scenarios caused by threat sources, vulnerabilities exploited by the threat sources, and unwanted incidents caused by the threat sources as a result of exploiting vulnerabilities. The specific assets that are harmed by the unwanted incidents are also captured by the CORAS threat models. This corresponds to Step 5 of the CORAS method. Having created

CORAS threat models, we then identify as many relevant indicators as possible for each element in the threat model. Identifying indicators is an extension to the original CORAS method and is performed to facilitate a more dynamic risk assessment.

Indicators can be identified during all eight stages of the CORAS process. As the creation of CORAS models is based on collaboration, new information and, as such, possible indicators can be discovered at any stage. However, some stages are more relevant. Context establishment in addition to risk identification are particularly useful, as this is where system information is gathered and the threat models are created. After having completed the risk identification, most possible indicators will likely have been identified. The formal modelling of indicators is done after the risk identification is completed, which means in between steps 5 and 6 of the CORAS method, and is performed in a similar manner to risk identification. With a complete threat diagram and an overall view of the system, the risk assessor will be aware of the likely threat paths, components and vulnerabilities in the system, and can target more specific parts of it to identify possible indicators.

Some care must be taken to separate vulnerabilities and indicators, as overlap may occur. In many cases, a vulnerability can be more usefully presented with some uncertainty or probability attached. Here, an indicator may better represent inexplicit vulnerabilities than the vulnerability element. However, in other cases, indicators can be phrased so that the answer may be represented by the presence or absence of a vulnerability. In such cases, the risk assessor must exhibit judgment as to how likely that is to change. Change in the state of a system is more easily modelled with indicators than with more static vulnerability elements. For visual clarity, there is also the possibility of adding a vulnerability with an attached indicator. This will make the possible vulnerability explicit to the reader while still benefiting from the risk calculation provided by the indicators.

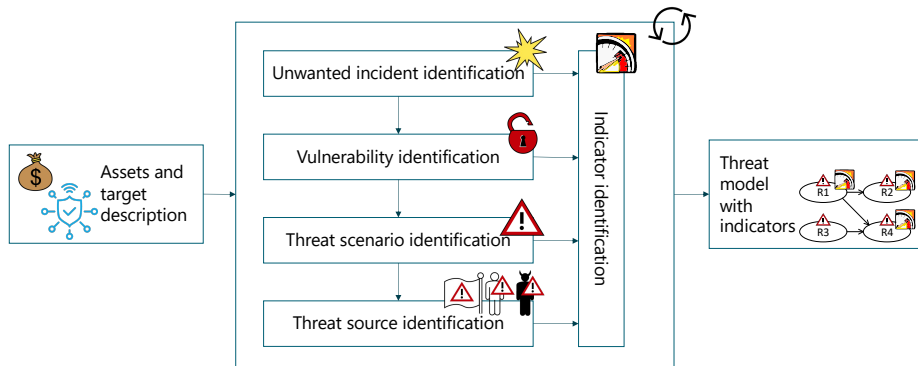


Fig. 2: Indicator identification method high level

4.2 Indicator translation

Indicator translation is the process of converting data from various indicator sources into a value that can modify the final risk value. The CORAS calculus explains how the final risk value is calculated. Provided the model is complete, the likelihood of an element can be determined based on its incoming relationships or links. There are established rules for working with both probabilities and frequencies, based on the user's preference. In this context, frequencies refer to how often an event occurs or is expected to occur.

The calculation is designed to work with value ranges, meaning the risk assessor can, for instance, set the category 'Rare' to a range of frequencies that do not overlap with the next level on the scale. These ranges can be defined according to the context of the case and do not need to be uniform. In other words, one range might cover years, while another might be measured in days.

The rules for determining likelihood differ based on whether the incoming links are from mutually exclusive or statistically independent elements. In most cases, in a regular diagram, the majority of elements stemming from a single preceding element can be considered statistically independent. These calculations are performed across the entire model to determine the likelihood of the unwanted incident, and thus the final risk value.

The calculus will be extended to account for the indicators. This means defining the allowed types of indicator values and how they impact the likelihood, conditional probability, or consequence of the elements they affect. How multiple indicators interact with one another will also be defined, to account for instances where there are many indicators affecting a single element and for instances where some indicators may be mutually exclusive.

Indicator sources present data in a variety of formats, ranging from simple booleans indicating presence or absence or integers counting detected events to more ambiguous data where a domain expert estimates policy quality. Certain tools may also present large amounts of data that must be parsed to represent a quantifiable risk-modifying value. For this reason, guidelines must be established to translate these heterogeneous sources into quantified metrics.

At this point, three valid formats have been identified: booleans representing the presence or absence of an indicator, integers representing the number of occurrences of an event, and integers representing the indicator data as a level on a scale.

Boolean indicators are the most common format because many questions can be phrased this way. Technical indicators can be on or off, indicating whether an occurrence or event is registered. In addition, for more social indicators, the risk assessor can remove ambiguity by rephrasing the indicator question. For example, instead of asking the case study owner to grade their access control policy, they could ask if it meets a certain standard. Boolean indicators are also well suited for categories related to test results and monitoring, where a positive test result or an event from monitoring software can be easily represented in this manner.

For indicators represented by the number of occurrences, the most common use case is application or network monitoring that records and logs events. A certain number of occurrences could indicate unusual or malicious behaviour, but in some cases, this threshold is not well represented by a Boolean yes or no. Increased or decreased activity can be used as indicators to gradually and subtly adjust the level of risk faced by the system. A system administrator could present a numerical metric that can be implemented as an indicator without defining an absolute threshold where an attack would be suspected.

Finally, some indicators cannot be represented as a boolean or a raw numeric metric. For example, an indicator representing the privacy risk inherent in a machine learning model should be rated based on how much data is leaked. In this case, there is no reasonable way to measure the number of occurrences, and setting a Boolean value would omit valuable information. For these cases, a scale system can be used in which the case study owner ranks severity on a scale. These scales can be of any granularity. For example, a risk assessment tool can provide a numerical metric from one to one hundred, which can be used to inform an indicator. Note that some effort must be made to ensure that the scale remains relatively linear to reduce ambiguity.

For all indicators, a certain level of user control is necessary. Attempting to capture the complexities of a system with simple, universally applicable quantifiable standards is unrealistic at this level of abstraction. Therefore, both the translation and calculation must be modifiable at the implementation level. This means that while reasonable defaults are set, users should be able to modify them if necessary. This could involve changing the number of levels on a scale, adjusting how much each level affects the calculation, or modifying the impact of an indicator value being on or off. Often the presence of an event carries more weight than its absence, though the opposite could also be the case.

For Boolean indicators, the user can determine how much the likelihood or consequence of the connected element is altered by a positive or negative result. This is represented by a float, by which the value is then multiplied. For numerical indicators, the user must set the maximum expected impact themselves, while the default remains as no change for no detected occurrences. In the case of scales, defaults are set for common scales, with the user able to modify the minimum and maximum impact.

Finally, it is noteworthy that most indicators can change format by rephrasing the question. Instead of implementing an indicator as a number of occurrences, it can simply ask if the number is above or below a certain threshold. Care should therefore be taken to ensure that the indicator question can be answered in the format most beneficial to the creator of the risk model.

4.3 Case studies

Our method has been evaluated in four case studies carried out within the context of the EU project NEMECYS [12], covering both the deployment and operation phases of connected medical devices. These case studies span diverse healthcare scenarios, including home dialysis treatment, wearable devices for

continuous monitoring of movement disorders, software as a medical device, and hospital-based point-of-care testing.

Due to time constraints, the full CORAS risk assessment process was condensed. The context establishment phase relied primarily on interview transcripts and supporting text documentation provided by each case study partner. Three meetings were held for each case study: (1) to confirm the initial high-level risk table, (2) to co-develop the base threat model, and (3) to identify relevant cyber-risk indicators. Preliminary models and tables were prepared in advance and refined collaboratively in each session. While the estimation, evaluation, and treatment steps are not yet completed, this phase delivered essential insight into relevant threats, vulnerabilities, and indicators in each context.

For each case study, we created a high-level risk table capturing potential threat sources, threat scenarios, vulnerabilities, unwanted incidents, assets at risk, and relevant components in the system under analysis. Section 5 provides a summary of the case studies, including the risks identified in each.

5 Results

In this section we first provide a summary of the case studies, then it shows an example CORAS threat model with indicators, followed by an overview of the number of indicators we identified for each indicator type. The reader is referred to our public deliverable [11] for a more detailed explanation of each risk indicator.

Case Study 1 - home dialysis treatment: This case study explores the cybersecurity implications of using a wearable sensor patch developed for real-time and continuous hydration monitoring in home dialysis scenarios. The technology aims to expand access to home dialysis for older patients by enabling remote clinical decision support. The primary cybersecurity risks identified include unauthorized modification or extraction of sensitive sensor data by hackers or malicious insiders, through methods such as tampering with firmware, exploiting debug ports, or performing adversary-in-the-middle (AiTM) attacks. These incidents could compromise the confidentiality, integrity, or availability of patient data, potentially leading to incorrect clinical decisions and health deterioration due to fluid imbalance. Additional risks include the theft of proprietary algorithms, which could result in financial loss and reputation damage. The main vulnerabilities are related to insufficient physical security, weak encryption, lack of secure communication protocols, and inadequate detection mechanisms.

Case Study 2 - wearable devices for continuous monitoring of movement disorders: This case study focuses on a wearable device system designed to provide continuous and objective monitoring of motor symptoms in patients with Parkinson's disease, improving treatment beyond traditional clinical exams. The system collects sensitive personal and clinical data using sensors and cloud-based infrastructure. The primary cybersecurity risks identified include unauthorized access to raw or processed patient data via insecure cloud services, physical access to local storage, or side-channel attacks on networked components. Potential threat

actors include hackers, insiders, and employees, who may exploit vulnerabilities such as a lack of encryption, poor access controls, static passwords, and insufficient physical security. Critical risks include breaches of patient confidentiality, loss of data integrity and availability due to configuration errors or malicious tampering, and extraction of proprietary algorithms, all of which could result in regulatory penalties and reduced trust in the system. The main components at risk include the monitoring devices, a smartbox, a physician portal, and Azure cloud services.

Case Study 3 - software as a medical device: This case study investigates mobile applications used by diabetes patients for therapy management, such as calculating carbohydrate intake and insulin dosage. Classified under the EU Medical Device Regulation (2017/745) [7] as Software as a Medical Device (SaMD), these apps must comply with stringent risk assessment requirements. The primary cybersecurity risks stem from their operation within general-purpose mobile environments, where vulnerabilities may be introduced through interactions with other applications, insecure device configurations, or weak communication protocols. Identified threats include adversary-in-the-middle (AiTM) attacks, data modification or extraction by hackers, technical malfunctions of external services, malicious code injection, and user error. These risks threaten patient data confidentiality, integrity, and availability, and in some cases directly harm patient health. Case Study 3 also highlights risks to the manufacturer's finances and reputation due to potential data breaches and non-compliance with data protection regulations. Key components at risk include the SaMD mobile app, NFC interfaces, mobile operating environment, external services, and web-based physician interfaces.

Case Study 4 - hospital-based point-of-care testing: This case study examines cybersecurity risks associated with self-testing using in-vitro diagnostic (IVD) medical devices, particularly for patients managing long-term conditions from home. While self-testing improves patient autonomy, reduces healthcare costs, and alleviates pressure on hospital resources, it also introduces new vulnerabilities. The primary risks identified involve unauthorized access to patient data across various components, including mobile apps, clinical portals, and the devices themselves (e.g., readers and sensors). Threat actors may exploit weak session management, improper authorization, insecure physical environments, and limited user security awareness to access, extract, or manipulate sensitive patient information. Additional concerns include malware injection through compromised systems, sensor data manipulation affecting clinical decisions, and broader reputation and financial risks to healthcare providers. The main assets at risk include patient data, sensor integrity, hospital systems, and the overall trustworthiness of the self-testing infrastructure.

5.1 Threat model

Figure 3 shows an excerpt from the model derived from case study 2. The scenario considers the risk of patient data becoming unavailable due to the edge device, called a SmartBox, malfunctioning. The Smartbox is responsible for collecting,

processing and transferring data to the system cloud servers. A lack availability of data can cause harm to the patient via improper or delayed treatment. In this part of the model three possible paths were identified, namely intentional and unintentional misconfiguration as well as an adversary getting access to the configuration web portal and altering the settings. The three adversaries are respectively *TS4: Employee*, *TS5: Insider* and *T6S: Hacker*, and the asset in question is *A1: Availability of data*.

For the unintentional misconfiguration path, we see it initiated by *TS4: Employee*, with the vulnerability *Lack of training* affecting the likelihood of initiation. To inform this vulnerability we use the indicator *IN47: How comprehensive is employee training?*. This leads to the threat scenario *T4: Unintentionally misconfigured Smartbox*, which propagates to threat scenario *T7: Basic device settings deleted or corrupted*.

For intentional misconfiguration the threat source is *TS5: Insider*, which initiates the threat scenario *T5: Intentionally misconfigured Smartbox*. The likelihood of this threat scenario is informed by the indicator *IN15: Are there screening procedures in place?* From here we propagate to threat scenario *T7: Basic device settings deleted or corrupted*.

The final path representing an external adversary is initiated by threat source *TS6: Hacker*. The web portal in question is on the local network, and requires close proximity to the device, so we inform the vulnerability *Lacking awareness training* with the indicators *IN26: How comprehensive is the awareness training?* and *IN10: Is the access policy too broad allowing access from unauthorized personnel?*, in addition to *IN34: How many employees clicked on a phishing link?* to approximate a value for this vulnerability. In addition, we add the vulnerability *CWE-308: Use of Single-factor authentication* which is informed by the indicator *IN48: Does the website use multi-factor authentication?* This leads to the threat scenario *T6: Adversary gets access to the web portal* where the likelihood is affected by the indicator *IN7: How many connection attempts does the portal receive per minute?*. Finally, this propagates to threat scenario *T7: Basic device settings deleted or corrupted*.

All these lead to the threat scenario *T7: Basic device settings deleted or corrupted*, where the edge device will attempt an online backup of its device settings. At this point, we note that a poor internet connectivity could interfere with the planned backup. To cover part of this, we include the vulnerability *Poor internet connectivity* and attach the indicators *IN6: Are there frequent blackouts due to weather and infrastructure?* and *IN5: How reliable is the internet connectivity?* to inform this vulnerability. Should the backup fail we denote this with threat scenario *T8: Device backup fails*, which results in the unwanted incident *U2: Device malfunction/shutdown* harming the asset *A1: Availability of data*.

The indicators in Figure 1 fall into the following categories: *IN6*, *IN10*, *IN15*, *IN25*, *IN26*, *IN47* and *IN48* are of type **Expert knowledge**, *IN5* and *IN34* are **Test results**, while *IN7* and *IN4* are of type **Application monitoring** and **Network monitoring** respectively.

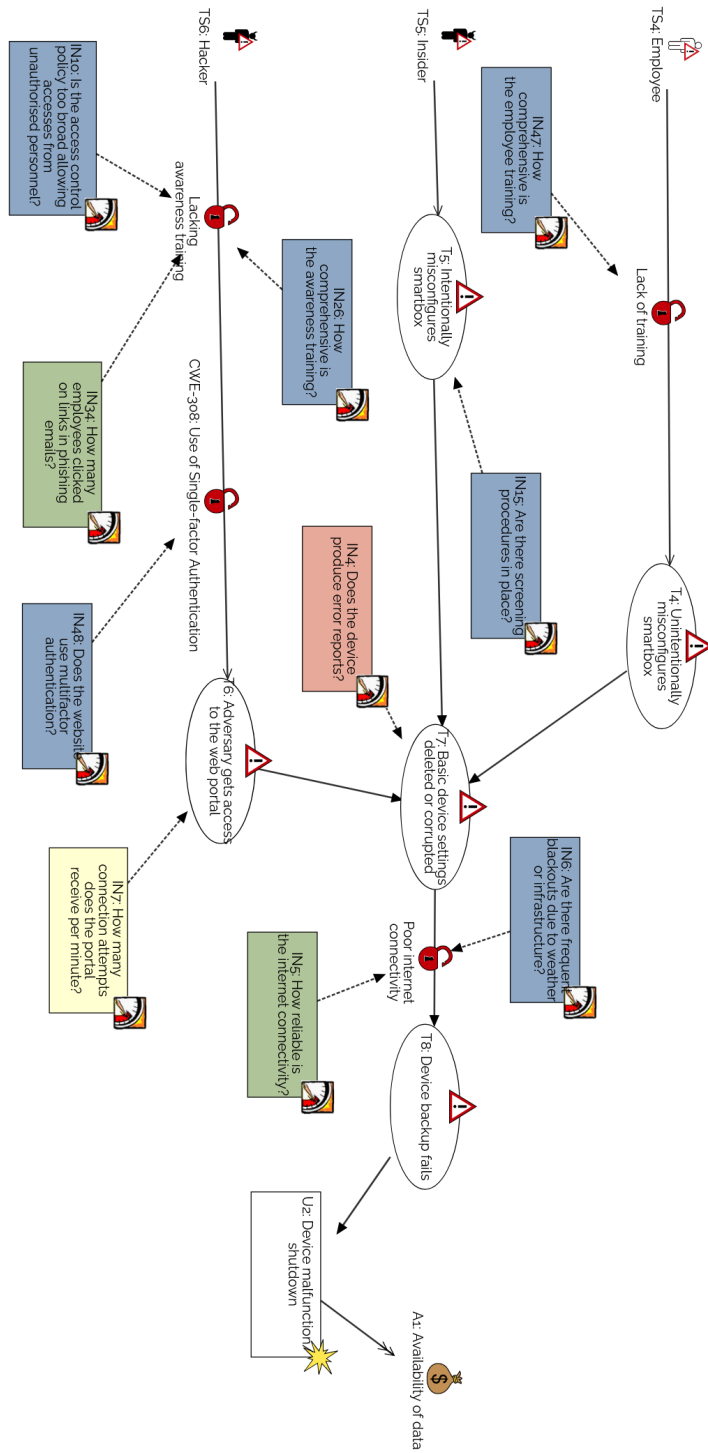


Fig. 3: Threat model

5.2 Indicators

The aggregated results of the risk identification for all case studies are presented in Table 6. The final count of indicator instances identified during the project totals 140, of which the majority are of type *expert knowledge* and *test result* compared to the more technical categories. Some of these are repeated, and the final count of *unique* indicators is 106. The overlap is present mostly in the expert knowledge category, as several indicators are sourced from more general knowledge of a system, though the test results category have some overlap as well. Due to the difference between the case studies there is no overlap in the network and application monitoring categories.

Table 6: Indicator count per type

Case study	Expert knowledge	Test Result	Network monitoring	Application monitoring
1	12	13	2	10
2	17	11	2	6
3	24	10	3	3
4	16	7	0	4
Total	69	41	7	23

To show how the format presented in Table 1 is used, we present another identified example indicator relevant to case study 4, which concerns itself with the transfer of data from the patient device to the physician system. A risk was identified were either a malicious insider or an adversary received access to and inserted a compromised reader or a device falsified to look as such into the hospital system. In this case a possible indicator was identified concerning the physician’s knowledge of legitimate devices. This indicator is presented below in Table 7

5.3 Discussion

Due to the range of defined indicators and the breadth in each category, there are many possible sources of indicators. One such source, as presented in Section 2, is the output of technical tools, which could be application or network monitoring tools, or tools used as part of a penetration test. While those indicators are less populous in the findings, they are still present. Most prevalent are the risks related to expert knowledge. One example is the indicator presented in Table 7. Another is *Access control policy* which has a substantial impact on physical security and can vary drastically within each use case. The distribution of identified indicators is affected by several factors. One of these is the lack of a functioning testbed from which to extract information. As a result, some

Table 7: IN66: Staff Awareness of Permitted Readers

ID:	IN 66	
Question:	IN66: Are the staff aware of all permitted readers?	
Data type:	Boolean	
Motivation:	The blood glucose readers transfer data to the system’s computer via a USB connection, which can be exploited to infect the system with malware. To mitigate this risk, the staff responsible should be made aware of all permitted readers to avoid inserting a malicious device into the hospital system.	
Indicator type:	Expert knowledge	X
	Test results	
	Network-layer monitoring	
	Application-layer monitoring	
Means of obtaining indicator value:	Consult policy if possible and interview staff regarding adherence and awareness.	

possible technical indicators were omitted due to a lack of data, and what could otherwise have been several separate indicators were aggregated into one indicator at a higher level of abstraction. In addition, as the process was performed in less time than recommended and with less case partner interaction, more generic indicators were often preferred, and thus the resulting models and indicators can still be expanded upon.

5.4 Lessons learned

All case study owners found the risk assessment process useful, with reports that the threat models and indicators contributed to a clearer and more significant understanding of the risk picture they are exposed to. Indicators were found to be a relevant addition to the models for their respective use cases, and in some cases simplified stakeholder communication. As indicators are phrased as simple questions, it can make it easier for stakeholders with a less technical background to contribute to the discussion. Asking specific yes or no questions allowed for direct answers, or if none were available, pointed out areas of the system where more attention was needed.

It can be challenging to choose the right level of technical abstraction for each indicator. General indicators can be more easily reused across case studies, and will often be easier to understand when reading the model. On the other hand they will often be less useful to each specific case, in particular in the case of technical indicators concerning bespoke system components. For the majority of the indicators, we made them as specific as information allowed, while restricting them to two or three per relation to avoid information overload. This approach resulted in many indicators which were reusable across all case studies, and the enumeration of indicators listed in deliverable [11] simplified the process

of indicator identification for the following case studies. It can with benefit be extended and refined further to facilitate future risk assessments.

6 Conclusion

In conclusion, indicators show promise in providing a flexible and adaptable means of accounting for current contextual information when modelling actual risk values. Building on the process of CORAS modelling, our established method for deriving these indicators has shown to be effective, particularly through collaborative brainstorming sessions with our industry case study partners. These sessions have enabled us to generate useful and relevant indicators that can be applied in real-world scenarios.

Moreover, we have developed practical guidelines to translate data from heterogeneous sources into simple risk value modifiers. This capability allows us to handle diverse data inputs and convert them into meaningful risk assessments. However, it is important to note that further work is required to refine these translation processes and to verify the overall usefulness and accuracy of the indicators we derive.

6.1 Future work

Future work will include further elaboration on the various indicator formats and data types, in addition to specification of the indicator calculus. Besides, more indicators and indicator types should be sourced from real tools to determine their feasibility and usefulness. By incorporating a wider range of indicators from practical applications, we can better assess their relevance and effectiveness.

6.2 Threats to validity

The indicators have been identified during a shortened version of the CORAS method, with less time spent with the case partners. This will inevitably affect the number and detail of the indicators identified. In addition, the lack of an accessible testbed results in an increased proportion of less technical indicator types, in addition to the fact that technical indicators identified are presented at a higher level of abstraction.

References

1. Adaros-Boye, C., Kearney, P., Josephs, M., Ulmer, H.: An indicators-of-risk library for industrial network security. In: Proceedings of the 16th International Conference on Availability, Reliability and Security. ARES '21, Association for Computing Machinery, New York, NY, USA (2021). <https://doi.org/10.1145/3465481.3470023>
2. Asiri, M., Saxena, N., Gjomemo, R., Burnap, P.: Understanding indicators of compromise against cyber-attacks in industrial control systems: A security perspective. *ACM Trans. Cyber-Phys. Syst.* **7**(2) (Apr 2023). <https://doi.org/10.1145/3587255>

3. Bernsmed, K., Jaatun, M.G.: Security-by-design challenges for medical device manufacturers. In: Proceedings of the 2024 European Interdisciplinary Cybersecurity Conference. p. 155–160. EICC '24, Association for Computing Machinery, New York, NY, USA (2024). <https://doi.org/10.1145/3655693.3661297>
4. Erdogan, G., Carmichael, L., Taylor, S., Tverdal, S., Skytterholm, A.N.: Dynamic cyber risk assessment for connected medical devices: the nemecys approach. In: Joint Proceedings of RCIS 2024 Workshops and Research Projects Track co-located with the 18th International Conference on Research Challenges in Information Science (RCIS 2024). p. 3674 (2024), <https://eur-ws.org/Vol-3674/RP-paper6.pdf>
5. Erdogan, G., Gonzalez, A., Refsdal, A., Seehusen, F.: A method for developing algorithms for assessing cyber-risk cost. In: 2017 IEEE International Conference on Software Quality, Reliability and Security (QRS). pp. 192–199 (2017). <https://doi.org/10.1109/QRS.2017.29>
6. Erdogan, G., Refsdal, A.: A method for developing qualitative security risk assessment algorithms. In: Cuppens, N., Cuppens, F., Lanet, J.L., Legay, A., Garcia-Alfaro, J. (eds.) Risks and Security of Internet and Systems. pp. 244–259. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-76687-4_17
7. European Union: Regulation (eu) 2017/745 of the european parliament and of the council of 5 april 2017 on medical devices, amending directive 2001/83/ec, regulation (ec) no 178/2002 and regulation (ec) no 1223/2009 and repealing council directives 90/385/eec and 93/42/eec (2025), <https://eur-lex.europa.eu/eli/reg/2017/745/oj/eng>
8. Lund, M.S., Solhaug, B., Stølen, K.: Model-driven risk analysis: the CORAS approach. Springer Science & Business Media (2010). <https://doi.org/10.1007/978-3-642-12323-8>
9. Meland, P.H., Tokas, S., Erdogan, G., Bernsmed, K., Omerovic, A.: A systematic mapping study on cyber security indicator data. *Electronics* **10**(9), 1092 (2021). <https://doi.org/10.3390/electronics10091092>
10. MITRE: MITRE ATT&CK (2025), <https://attack.mitre.org/>
11. NEMECYS: Deliverable D2.1 – Risk Benefit Schemes (initial) (2025), <https://nemecys.eu/resources/public-deliverables/>
12. NEMECYS: NEMECYS - New Medical Cybersecurity Assessment and Design Solutions (2025), <https://nemecys.eu/>
13. Silvestri, S., Islam, S., Amelin, D., Weiler, G., Papastergiou, S., Ciampi, M.: Cyber threat assessment and management for securing healthcare ecosystems using natural language processing. *International Journal of Information Security* **23**(1), 31–50 (2024). <https://doi.org/10.1007/s10207-023-00769-w>