

Politecnico di Torino

ScuDo
Scuola di Dottorato ~ Doctoral School
WHAT YOU ARE, TAKES YOU FAR

Doctoral Dissertation
Doctoral Program in Computer and Systems Engineering (35^{th}cycle)

Methodologies and tools to support Vulnerability Assessment and Cyber Risk Analysis activities within the Italian Cybersecurity Perimeter

By

Nicolò Maunero

******

Supervisor(s):
Prof. Paolo Prinetto, Supervisor

Doctoral Examination Committee:
Prof. Alessandro Armando, Referee, Università degli Studi di Genova
Prof. Francesco Buccafurri, Referee, Università Mediterranea di Reggio Calabria
Prof. Roberto Baldoni, Agenzia per la Cybersicurezza Nazionale
Dr. Fabio De Rosa, CINI Cybersecurity National Lab.
Prof. Tiziana Margaria, University of Limerick
Prof. Matteo Sonza Reorda, Politecnico di Torino

Politecnico di Torino
2023

Declaration

I hereby declare that the contents and organization of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

Nicolò Maunero
2023

* This dissertation is presented in partial fulfillment of the requirements for **Ph.D. degree** in the Graduate School of Politecnico di Torino (ScuDo).

Ai miei gentitori, ai miei fratelli, alla mia famiglia per l'incondizionato supporto.

Acknowledgements

Dedico questo spazio a tutte le persone che mi hanno accompaganto e supportato durante il percorso che ha portato alla stesura di questo elaborato, senza le quali nulla di tutto questo sarebbe stato possibile. Un ringraziamento particolare va al Professor Paolo Prinetto, mentore insostituibile che ha saputo guidarmi e consigliarmi saggiamente nelle scelte professionali e scientifiche, ma soprattuto persona che ha saputo trasmettermi valori umani che vanno ben oltre la sola sfera professionale. Ringrazio Fabio De Rosa che mi ha supportato e guidato in questo percorso di crescita professionale. Un sentito ringraziamento va agli studenti con cui ho avuto il piacere di collaborare, che con il loro lavoro di tesi hanno contribuito alla realizzazione di parti fondamentali di questo lavoro. Un ringraziamento speciale va a Gianluca Roascio amico e collega con il quale ho condiviso, fin dall'inizio, questo percorso, nei suoi momenti positivi e negativi. Inoltre tengo a ringraziare in particolare anche Martina Trussoni e Federico Talentino il cui aiuto e supporto è stato particolarmente prezioso nella realizzazione di questo lavoro. Infine un ringraziamento è dovuto a tutti i colleghi con cui ho avuto il piacere di collaborare in questi anni per il supporto e le opportunità di crescita offerte.

Abstract

In the last decade, we have witnessed a steady increase in cyber attacks to the detriment not only of large organizations but also of small and medium-sized businesses as well as individuals. These attacks range from denial of service of provided services to data theft or destructive actions of malware. The constant presence of this menace makes it necessary and essential to set up actions to analyze the security posture of information systems, so that cyber attacks can be prevented or mitigated.

The process of *risk assessment* can be conceptually divided into 3 main phases: (i) assessment of the threats to which a computer system may be subjected, also known as *threat modeling*; (ii) assessment of the vulnerabilities present in the system, also known as *Vulnerability Assessment and Penetration Testing* (VAPT); (iii) finally the *risk evaluation* that combines the information obtained from the previous two phases with information related to the specific case under analysis (such as the impact that the implementation of a threat may have on the system). The output of this entire process aims to have a quantitative assessment of the security posture of the analyzed ICT infrastructure.

The main problem with these types of activities is that, to date, they are still performed predominantly manually and, therefore, require analysts a great deal of experience and precision to achieve significant results. In addition, the constant increase, on the one hand of new threats that are discovered every day, and, on the other hand, of the complexity of ICT infrastructures adopted even by small organizations, leads to a non-trivial complexity in the management of the large amount of security-related information items. Therefore, it becomes important to introduce some degree of automation in risk assessment activities, to support and simplify the implementation process and the management of security data. Automating, however, brings in challenges and specific requirements, including, among the others: (i) the definition of a formal modeling language to represent the infrastructure under analysis and the related security information; (ii) the identification of a set of formal rules for the automatic and effective inference of the necessary information.

The goal of this thesis work is, therefore, to provide tools to support risk assessment activities by introducing automation in different phases of the security assessment procedure. To achieve this, an ontology has been firstly introduced supporting the definition of a formal and semantic representation of the data needed for the analysis and a representative metamodel of a generic ICT infrastructure. The defined ontology is properly enriched with information useful for effectively representing system threats and vulnerabilities.

Considering the various stages of risk assessment, approaches have been designed to automate the collection of information related to known infrastructure vulnerabilities. Starting from the list of assets present, the ontology can be automatically populated with information related to known vulnerabilities for those assets, gathered from external sources.

Concerning the threat modeling process, formal rules have been defined: interpreted by ontology automatic reasoners, these allow the automatic derivation of the threat modeling of the target infrastructure, identifying the potential threats of each involved asset.

Finally, this information is combined with the available information on possible business impact of the occurrence of a threat and its occurrence probability, to propose a qualitative cyber risk assessment for the identified threats.

The proposed solution follows an asset-oriented approach in defining the ontology meta-model; this allows to tightly link together infrastructure components and security data to facilitate the integration of automation and obtain better and more precise results.

The benefit of such a solution is that it can simplify and speed up the execution of risk assessment activities, allowing for great adaptability in an ever-changing landscape.

Contents

List of Figures x

List of Tables xii

1 Introduction 1

2 Motivations 8

2.1 Italian National Cybersecurity Perimeter 9

3 Background 13

3.1 Cybersecurity Frameworks & Risk Assessment 13

3.1.1 NIST Cybersecurity Framework 16

3.1.2 Italian Cybersecurity Framework 28

3.2 EU Risk Assessment Framework 37

3.3 Threat Modeling Methodologies 43

3.3.1 STRIDE 45

3.3.2 Attack Trees 47

3.3.3 LINDDUN 47

4 State-of-the-Art 51

4.1 Cybersecurity Ontologies 52

4.1.1 Unified Cybersecurity Ontology 52

4.1.2 IoTSec Ontology 54
4.2 Vulnerability Management 55
4.2.1 Ontology-based Solutions 56
4.3 Threat Modeling Automation 60
4.3.1 General Solutions 60
4.3.2 Ontology-based Solutions 61
4.4 Risk Assessment 64
4.4.1 General Solutions 64
4.4.2 Ontology-based Solutions 67
4.4.3 Considerations 70

5 Proposed Solution **72**
5.1 Architecture 72
5.2 Ontology 79
5.2.1 Modeling ICT Infrastructures 85
5.2.2 Modeling Supply-chain 88
5.2.3 Modeling Data Flow Diagram 89
5.2.4 Vulnerabilities 90
5.2.5 Threats 94
5.2.6 Risk 98
5.3 Vulnerability Management: Proposed Approach 100
5.4 Threat Modeling: Proposed Approach 104
5.4.1 Threat Modeling Logic 106
5.5 Risk Assessment: Proposed Approach 109
5.5.1 Risk Evaluation 111

6 Results **116**
6.1 Preparing the *Cybersecurity Knowledge-base* 116

6.2 Performing Risk Assessment 122
6.2.1 Considerations 127

7 Conclusions **129**

References **133**

List of Figures

3.1 NIST Proposed Risk Model [1] 27

5.1 High Level View of the Proposed Solution 73
5.2 General Architecture of the Proposed Approach 74
5.3 Sequence Diagram of Proposed Approach Activities 77
5.4 Complete View of the Proposed Ontology 84
5.5 Ontology for Modeling ICT Infrastructures 85
5.6 Ontology for Modeling ICT Infrastructures Supply Chain 89
5.7 Ontology for Modeling ICT Infrastructures Data Flow Diagram 89
5.8 Ontology for Modeling Vulnerabilities 91
5.9 Ontology for Modeling Threats 95
5.10 Ontology for Modeling Threats 99
5.11 Architecture of the Proposed Vulnerability Management Approach 101
5.12 Sequence Diagram of Activities Performed for Vulnerabilities Management . 101
5.13 Architecture of the Proposed Threat Modeling Approach (ThreMA) [2] . . . 104
5.14 Sequence Diagram of Activities Performed for Threat Modeling [2] 105
5.15 Architecture of the Proposed Risk Assessment Approach 109
5.16 Sequence Diagram of Activities Performed for Risk Assessment 110

6.1 Use-case Infrastructure 118
6.2 Information for Risk Assessment identified for component *OS1* 123

6.3 Information for Risk Assessment identified for component *APP5* 124

6.4 Information for Risk Assessment identified for component *FIRM1* 125

6.5 Example of computed risk for *threat-component* pairs 126

List of Tables

5.1 Base Metrics CVSS Scores 93

5.2 Threat Categories and Associated Threats 98

5.3 Inference Rules 108

5.4 Risk Factor Conversion Scale 113

5.5 Risk Evaluation Conversion Scale 114

6.1 Threat Modeling Results 117

6.2 Number or extracted vulnerabilities for given products 120

6.3 Example of extracted vulnerabilities information for Component OS1 121

6.4 Example of extracted vulnerabilities information for component APP5 121

6.5 Example of extracted vulnerabilities information for component FIRM1 . . . 122

Chapter 1

Introduction

Recent years have seen a continuous increase in cyber attacks, accentuated even more by the COVID-19 pandemic and the associated remote working [3]. The targets of the attacks are the most disparate, from the single individual to large companies and public organizations, contributing to an annual cost, in terms of damage, of several billion [4] (e.g., Wannacry alone was estimated in costing to affected companies more than 4 billion dollars in total [5]). Therefore, it is becoming more and more important to put in place adequate defenses to tackle an ever-evolving world.

Given the steady increase in cybercrime, governments in various countries have been prioritizing and strengthening investment plans to improve the security posture of the country's infrastructures, with particular emphasis on critical infrastructure protection. Notable examples are the 2017 Executive Order of the US President [6] and the European Union's NIS Directive, now at its second iteration [7].

The need to guarantee predetermined levels of security and the constant growth of the complexity of systems and digital infrastructures makes it necessary, hence, to adopt tools and solutions to support the management and governance of this information; security teams need to manage and correlate security data (i.e., vulnerabilities and threats) and the composition of the infrastructure under analysis to identify vulnerable points and plan an effective cybersecurity risk management strategy.

Cybersecurity Risk Management, similarly to its business counterpart, is the process of forecasting and evaluating the risk related to cyber attacks and incidents, together with the procedures to mitigate or avoid their impact [8].

In the various proposed Risk Management frameworks [9–12], one of the key parts of the whole process is the *Risk Assessment* [13]. It considers different phases of the system life cycle, from design to actual implementation, intending to assess the system cybersecurity posture and preparedness level against possible cyber attacks. The *Risk Assessment* process can be divided into three main phases:

- *Threat Modelling*, aimed at identifying the threats the system is exposed to, e.g., the possible attack vectors and how attackers could exploit any vulnerability or flaw in the system;
- *Vulnerability Assessment* (VA), which is the iterative process of identifying vulnerabilities, and
- *Penetration Testing* (PT), in which identified vulnerabilities and potential issues are thoroughly analyzed by trying to exploit them. These three phases synergistically compute the level of cyber risk to which the system is exposed.

Risk, and its contributing factors, can be assessed in a variety of ways, including *quantitatively*, *qualitatively*, or *semi-quantitatively* [1]. There is no one method that is more correct than the other, but the risk assessment process that an organization chooses to adopt must take into account several factors and how the organization intends to deal with risk, how it perceives it, and what communication strategy it intends to adopt.

In order to assess risk using numbers, quantitative assessments often follow a set of principles, rules, or methodologies. The best cost-benefit evaluations of potential risk responses or courses of action are supported by this kind of assessment. The interpretation and explanation of the quantitative results, particularly the assumptions and limitations on utilizing the results, may be necessary since the significance of the data is not always obvious. For instance, companies may frequently question the accuracy of the data or findings from risk assessments or the significance of variations in the results. Additionally, when subjective judgments are concealed inside quantitative evaluations or when there is a great deal of uncertainty around the calculation of values, the rigor of quantification is considerably reduced. The choice to use this type of approach depends on an assessment, and trade-off, between the costs and benefits it can bring. In fact, a qualitative risk analysis requires the use of advanced tools and the involvement of highly skilled personnel who are capable of dealing with such an analysis.

Rather than using numerical categories or levels to categorize risk (such as very low, low, moderate, high, or very high), qualitative assessments often use a set of methodologies, principles, or norms. This kind of evaluation encourages sharing risk findings with decision-

makers. However, the spectrum of values in qualitative evaluations is typically rather narrow, making it challenging to compare or order the stated dangers in terms of relative importance. Additionally, various experts depending on their unique expertise may generate noticeably varied evaluation outcomes unless each metric is extremely well defined or is supported by meaningful instances. With regard to a qualitative assessment, it is important to ensure the reproducibility of the results; this can be achieved by providing detailed and precise explanations of why a given metric was assigned a certain level but also by making use of tables for combining qualitative values and representation to ensure comparability of results.

Finally, a semi-quantitative approach attempts to combine the benefits of both other approaches. Numerical scales (e.g., 1-10) or a set of ranges of numerical values (e.g., 0-15, 16-30, and so on) are used to represent the data, allowing easy and effective representation of the results obtained. Furthermore, this allows, with respect to pure qualitative assessment, to communicate and better compare the results obtained; vine better convey the difference between the identified risk levels. In the semi-quantitative approach, the fundamental role that an expert assessment plays in assigning these values shines through even more. In addition, if the numerical scales used provide sufficient granularity in the representation of the data obtained, the order of priority among the results can be identified more accurately than with a purely qualitative approach. Each bin or range of values needs to be precisely defined and/or described by relevant examples; when subjective judgments are hidden within evaluations or when there is a lot of ambiguity surrounding a value decision, the rigor is greatly reduced, much like with a quantitative method.

The risk assessment process can be performed in a number of ways that vary with respect to the point of view an organization want to use and what aspects of risk they want to focus on. In addition, the approach used may vary depending on the level of detail to which one wants to go in the analysis, how one wants to compare and reuse the results obtained in scenarios similar to the one in which the analysis is carried out, and so on. Analysis approaches can be divided in the following categories: (i) *threat-oriented*, (ii) *asset/impact-oriented*, or (iii) *vulnerability-oriented* [1].

A threat-oriented strategy begins with the identification of threat sources and threat events, the goal is to identify possible threat scenarios and how these can develop throughout the organization. Given the identified threats, vulnerabilities are identified and an estimation of the possible impact (damage the realization of a threat can have on the organization) is performed.

An asset/impact oriented approach focuses first on identifying assets, and from these, assets considered critical to the organization are selected. This process is performed taking into account what are the business goals of the organization and an estimate of the impact resulting

from the compromise of critical assets is made. At this point, the factors that can lead to the identified consequences are analyzed and evaluated.

A vulnerability-oriented approach starts by analyzing the vulnerabilities in the system, considered as predisposing conditions that can lead to the realization of a threat. These threats are then identified and evaluated according to the impact and consequences they may have on the organization.

Each analysis method includes the same risk elements and, as a result, requires the same set of risk assessment tasks, but in a different sequence. The findings of the risk assessment may be affected as a result of different starting points, leaving certain threats undetected. Therefore, identifying risks from a different viewpoint can strengthen the analytical rigor and efficacy (for example, by combining a threat-oriented analysis technique with an asset/impact-oriented analysis approach).

In addition to the type of approach to analysis that an organization decides to follow, better and more accurate results can be achieved by adopting analytical methods that take into account the many relationships that exist among various risk factors. It should be considered that a threat may materialize from several conditions, or that one starting condition may lead to the realization of several threats; a vulnerability may be predisposing condition for one or more threats, as multiple vulnerabilities may be for a single threat. Thus, an effective risk analysis approach might take into account and mix together elements of the various possible approaches.

Risk assessment results are fundamental for an effective risk management process, in fact on them the strategy to improve the security of the system is built and defined. Threats that have been assigned a higher level of criticality must be addressed immediately, while it may be decided that threats with a criticality lower of a given threshold are acceptable risks. An organization can decide to adopt one of the various risk management strategies based on their business goals and on how they decide to manage the risk (e.g., manage it directly, transfer it to third parties, and so on).

In all of this, in recent years it has become increasingly clear how a formal and shareable, but above all machine-readable, language is necessary to adopt in the risk assessment process. This not only allows for a common medium with which to share results, interpret and compare them, but more importantly gives the possibility of using automation solutions in various parts of the assessment process. An explanatory example is NIST's OSCAL (Open Security Controls Assessment Language) project [14] still in development. This project is many-fold, including: (i) to address information security and privacy risks in the verification of security control implemented, (ii) to provide a way of describing and assessing assurance of a system's

security and privacy posture, (iii) to standardize the process of documenting, implementing and assessing security controls, and (iv) to enable automation in the whole process.

In the research literature, the same problem has been addressed over the years leveraging *ontologies* as the mean of formally representing information in a machine-readable format. The semantic representation of information [15] through ontologies is usually adopted for the creation of formal models by which concepts, even complex ones, can be represented and relationships can be defined, seeking to ensure unambiguous meaning.

Ontologies are viewed as the foundation of semantic inference, a means for managing knowledge, finding new linkages, and automatically analyzing the content of data. These inference-based methods are crucial for identifying any discrepancies in the data used. When there may be ambiguities in the words used in the various datasets, for instance, or when a little more information may lead to the identification of new linkages, ontologies play a role in assisting data integration. Ontologies are an explicit and formal method of representing concepts and their relationships to convey their meaning using classes, attributes, relationships, and axioms [16].

The representation of semantically rich information using ontologies allows for the inference of new knowledge [17], in terms of the relationships between the data used; ontologies allow for the analysis and management of the knowledge represented in them. Of course, the application of inference techniques also allows the identification of errors or inconsistencies in the data. Ontologies enable formalism through the use of a descriptive and representational language that is machine-readable. They allow for the explicit representation of information through the definition of classes, relationships, properties and rules. Finally, they allow the representation of data and information in a conceptual way through the definitions of high-level models of the knowledge domain represented.

The work of this Thesis goes toward the automation of the risk assessment process of a generic ICT infrastructure leveraging on an ontology as the *cybersecurity knowledge-base* to support the various phases of the process. The proposed solution is conceptually divided into four main part:

- **System Modeling**: provides the necessary vocabulary to model the ICT infrastructure under analysis.
- **Vulnerabilities Management**: provides the necessary vocabulary to model vulnerabilities of the systems, how they are related with system components and a way of semantically represent information for automated analysis and inference of knowledge.

- **Threat Modeling**: provides the necessary vocabulary to model threat that may affect an ICT Infrastructure, the threat catalogue (i.e., a list of reference threats) and the inference rules necessary to automate the process of identifying threats of the target infrastructure.
- **Risk Assessment**: provides the mean of combining threat and vulnerabilities information to evaluate associated risk and rank threats accordingly.

The *Threat Modelling* phase is of particular importance since it can be used to drive the entire risk assessment and management process, as well as to provide useful insight on the potential vulnerabilities in the infrastructure to be investigated by dedicated VAPT campaigns.

Since Threat modelling is still today a mainly manual activity [18], it requires highly trained personnel, and it is often error-prone [19]. In addition, these problems are compounded by the rapid and constant growth of ICT infrastructure complexity [20] and related cyber threats [21, 22]. Therefore, it becomes crucial to create solutions aiming to automate the Threat Modelling process, in particular, to develop tools able to prevent human errors and simplify the management of organizations' ICT infrastructures [23]. Moreover, automation enables a faster reaction to changes both in the infrastructure (e.g., technology, architecture) and in threats (e.g., new attack vectors), allowing the security operations team to plan remediation activities and improve the system security posture in a shorter time.

Over the years, several solutions trying to automate the threat modelling process have been proposed [24] in both academia and industry. Most of them are focused on specific vertical sectors, such as the world of IoT [25] or of Industrial Control Systems (ICS) [26], which makes them harder to be adopted to more general ICT infrastructure scenarios. On the other hand, "generic" approaches and solutions available in the literature often either consider threats in a too broader way or are too tied to known system vulnerabilities, thus risking resulting in ineffective system analysis [27]. Moreover, supply chain threats are often marginally considered [28].

Generally speaking, automating the threat modelling process brings with it two main challenges: the former one being the need for a standard representation of models and data used in various stages of the process, establishing a formal and common vocabulary for all the involved parties. Often, threat modelling is performed by different specialists, with different levels of security knowledge; hence, different interpretations of models and data may take place, thus generating misunderstanding and ineffectiveness during the overall threat modelling process [29]. The latter challenge is the definition of a well-defined set of inference rules supporting the reasoning task during the threat identification process. This set of rules has to be based on formal models and data representing the target ICT infrastructure. Frequently,

both these data and their threats lack the right context, thus leading to low accuracy in terms of domain knowledge compared with what a domain expert could do [30]. Such a problem can effectively be addressed by exploiting ontologies. These are often designed to support computation with structured data by providing a standardized descriptive vocabulary for concepts and machine-processable entity-relation semantics, thus, providing the possibility of performing automated reasoning to extract the desired information [30].

In general, the work presented in this Thesis leverages information from open source and publicly accessible cybersecurity sources. In the vulnerability management process, starting with information about infrastructure components, known vulnerabilities are identified from the CVE [31], CWE [32] (both maintained by MITRE [1]) and NVD [33] (maintained by NIST[2]) knowledge bases; this also provides information on the severity rating of these vulnerabilities (CVSS score) which are subsequently used for risk calculation. In the threat modeling process the possible threats are extracted from the CAPEC [34] knowledge-base (maintained by MITRE), this also allows access to a general qualitative assessment with respect to the possible impact and probability of occurrence of these threats; also this information is used in the risk calculation. Having this data available, the risk calculation can be done without virtually needing user intervention.

Although the proposed solution relies on publicly accessible information in an attempt to automate the entire risk management process as much as possible, it still remains an approach that can be used in general leveraging on the provided ontology as a means of information representation that can be populated with data collected from vulnerability assessment and threat modeling operations performed in different ways and with different methodologies. However, using the ontology for the semantic representation of this information allows the process of analyzing identified data and information to be automated.

The remainder of the document is structured as follows: Chapter 2 provides a brief overview of the motivations behind the Thesis work. Chapter 3 provides an overview of risk management and assessment frameworks, both at the international and Italian levels. Chapter 4 provides an overview of the academic research literature on the topics discussed in this Thesis. Chapter 5 provides a description of the implemented ontology and approach to the risk assessment process, while Chapter 6 presents the most meaningful results obtained. Finally, Chapter 7 draws the conclusions.

[1] https://www.mitre.org
[2] https://www.nist.gov

Chapter 2

Motivations

Following Article 5, comma 1, of Decree Law No. 82 of 14 June 2021 [35], converted with amendments by Law No. 109 of 4 August 2021, the Italian *Agenzia per la Cybersicurezza Nazionale* is established for the protection of national interests in the field of cybersecurity, with headquarters in Rome.

The Agency becomes the national authority for cybersecurity and the single contact point for network and information system security, for the purposes of the NIS Decree Law, so it assumes all the responsibilities the DIS had with respect to the implementation of the *Italian National Cybersecurity Perimeter* (PSNC) (more details in Section 2.1). The *PSNC* identifies essential service providers which shall comply with several obligations in terms of cybersecurity. One of the activities that should be performed is the periodic execution and maintenance of the list of assets of the ICT infrastructure; this information is collected by the *ACN*.

To deal with the collection of this information the *ACN* developed an **ontology** that is provided to essential service providers as a formalized and machine-readable way of collecting asset information.

It is precisely from the Italian legislative context that this thesis work developed. In fact, given ACN's development of an ontology for the collection of information regarding the assets of entities operating within the perimeter and the requirement for the same entities to perform periodic risk assessment activities it was decided to work on an ontology that would allow support in risk assessment.

Therefore, the idea was to link the two needs together and, on the one hand, develop an ontology that would allow the modeling of a generic ICT infrastructure with all the necessary details not only to collect the list of assets, but to enrich it with concepts that could be useful

for subsequent security analysis (i.e., the description of interactions between components). On the other hand, add to the ontological metamodel of the generic infrastructure all this security information (i.e., vulnerabilities and threats) necessary to provide a tool that could support an organization in performing the risk assessment process.

2.1 Italian National Cybersecurity Perimeter

With the Legislative Decree of 18 May 2018, n.65 [36], published on the *Gazzetta Ufficiale* n.132 of 9 June 2018, Italy implemented the *NIS Directive*, an EU-wise cybersecurity legislation [37], to define the measures necessary to achieve a high level of security of networks and information systems. The Order applies to the *Essential Services Operators* (OSE) and the *Digital Service Providers* (DSP):

- **OSE**: those entities, public or private, that provide essential services to society and economy in the fields of health, energy, transport, banking, financial market infrastructure, drinking water supply and distribution, and digital infrastructure;
- **DSP**: legal persons providing e-commerce, cloud computing or search engine services, with a principal place of business, registered office or designated representative on the national territory. The obligations for DSPs do not apply to companies which European legislation defines as 'small' and 'micro' companies, i.e., those which have fewer than 50 employees and an annual turnover or annual balance sheet not exceeding 10 million Euros.

According to the Legislative Decree, both the OSEs and the DSPs are required to adopt technical and organizational appropriate and proportionate measures to prevent and minimize the impact of incidents on the security of networks and information systems, to ensure continuity of service. In addition, they must notify, without undue delay, incidents that have a significant impact on service continuity and service provision respectively, to the Italian *Computer Security Incident Response Team* (CSIRT), also informing the relevant competent NIS Authority.

The tasks of the CSIRT are defined by Legislative Decree No. 65 of 18 May 2018 and the Decree of the President of the Council of Ministers of 8 August 2019 Article 4. They include [37]:

- defining procedures for the prevention and management of computer incidents;

- receiving incident notifications, informing the DIS, as a single point of contact and for the activities of prevention and preparation of possible crisis situations and activation of alert procedures entrusted to the Cyber Security Unit;
- providing the notifying party with information that may facilitate the effective management of the event;
- informing other EU Member States that may be affected by the incident, protecting the safety and commercial interests of the OSE or DSP as well as the confidentiality of the information provided;
- ensuring cooperation in the CSIRT network, by identifying forms of operational cooperation, exchange of information and sharing of best practices.

The establishment of the Italian *Perimetro di Sicurezza Nazionale Cibernetica* takes place within the framework of the NIS Directive.

The *PSNC* was established pursuant to Article 1, comma 1, of Decree Law No. 105 of 21 September 2019 [38], converted with amendments by Law No. 133 of 18 November 2019 (in the *Gazzetta Ufficiale* No. 272 of 20/11/2019) in order to ensure a high level of security of the networks, information systems and IT services of public administrations, public and private entities and operators having an office in the national territory, which provide an *essential service* and on which the exercise of *essential function of the State* depends.

The criteria by which essential service providers are identified are as follows:

- An entity, either public or private, exercises an *essential function of the State* where it is assigned tasks aimed at ensuring the continuity of the action of the Government and of the constitutional Organs, the internal and external security and defense of the State, international relations, security and public order, the administration of justice, the functionality of the economic and financial systems and transport.
- An entity, whether public or private, provides an *essential service* when it performs the following activities: activities necessary for the exercise of essential functions of the State; activities necessary to ensure continuity of supply and efficiency of infrastructure; and activities necessary to guarantee fundamental rights. Also considered are those research activities, in technology and productive sectors, which aim to ensure the strategic autonomy and development of the national economic system.

The above-mentioned entities and operators are found in different sectors: internal affairs, defense, space and aerospace energy, telecommunications, economy and finance, transport, digital services, critical technologies and social security/labour.

For each sector, the competent Minister must identify entities whose exercise of essential function or whose provision of essential service depends on networks, information systems and computer services whose malfunctioning, interruption, even partial, or improper use may result in a threat to national security. The result is a list of entities falling within the *PSNC*.

All of these subjects, according to the Decree, are required to prepare and update, at least once a year, a list of the networks, information systems and IT services for which they are responsible, including their architecture and components. They must, then, identify the ICT assets necessary to perform the essential function or service, in order to assess the impact of an incident on the ICT asset, in terms of its operability and the compromise of data availability, integrity or confidentiality (CIA triad) and assess dependencies with other networks, information systems, IT services or physical infrastructures pertaining to other entities. Finally, entities must identify the ICT assets that, in the event of an incident, would cause total disruption of the essential function or service.

The list compiled is transmitted to the Presidency of the Council of Ministers and to the Ministry of Economic Development and they will transmit it to the *Dipartimento delle Informazioni per la Sicurezza* (DIS) for the prevention, preparation and management of cyber crises entrusted to the Cyber Security Unit, as well as to the organ of the Ministry of the Interior for security. The transmission of ICT asset lists take place via a digital platform set up at the DIS.

The Decree of the President of the Council of Ministers (DPCM) of 15 June 2021 [39] identifies the categories of ICT assets, systems and services that must undergo an evaluation process when used by entities operating within the *PSNC*. Those categories are based on the technical criteria of the art. 13 of the Decree of the President of the Republic, no. 54, of 5 February 2021 [40], and are identified based on the functionalities they provide:

- technologies that provide switching capabilities in a network, enforcement of security policies, and identification of and protection from cyber threats in a network;
- technologies that provide command, control and implementation capabilities in an industrial control network;
- technologies that provide monitoring and configuration control capabilities of an electronic communication network;

- technologies that provide network security capabilities with respect to the availability, authenticity, integrity, or confidentiality of services offered or data stored, transmitted, or processed;
- technologies that provide authentication and resource allocation functionalities of an electronic communication network;
- implementation of a computer service through the use of an existing software program or the development, in part or in full, of a new software program which constitutes the application part relevant to the delivery of the computer service itself.

Based on these criteria, the implementing Decree established the categories referring to the hardware and software components that enable the implementation of these functions [41]. The following are the four identified categories:

1. Software and hardware components supporting network functionality and services (switching, transport, access control).
2. Hardware and software components that implement security features for networks and transmitted and processed data.
3. Hardware and software components that implement data acquisition functionalities, control and monitoring capabilities, and components that implement automation capabilities of industrial networks, infrastructure and systems.
4. Software components that implement security mechanisms.

A later Decree (DPCM n.81 [42]) further detailed the obligation of organizations operating within the *PSNC*, indicating which cybersecurity measures they have to comply with. These measures are the ones detailed in the Italian Cybersecurity Framework [43] and these include, among others, and thus are imposed: (i) the adoption and implementation of asset inventory and asset management procedures; (ii) the periodic execution of risk assessment operations which should consider not only “internal” risk, but also the risk related to the supply chain; (iii) the adoption of incident management and response procedure as well as the adoption of monitoring systems; and so on.

Chapter 3

Background

In this chapter some background concepts related to *risk management* and *risk assessment* will be presented. First, an overview of some Cybersecurity Frameworks and the different approaches they consider regarding the assessment of cybersecurity risk will be presented. After that, an overview of the main *threat modeling methodologies* will be provided, an aspect that plays a fundamental role in the risk assessment process. Regarding threat modeling, several possible approaches will be presented, each considering a different methodology or focusing more on particular aspects characterizing a cybersecurity threat.

3.1 Cybersecurity Frameworks & Risk Assessment

The primary goal of cybersecurity is to protect an organization or company from the threats posed by malicious actors who aim to compromise information systems by leveraging (exploiting) vulnerabilities or security flaws in systems or processes. Thus, it is clear that some level of risk is present from the possible compromise of systems. There are different definitions of risk, but in general it can be defined as the possibility of losing something valuable, value can be assessed in different ways: such as economic value, material possession, level of economic security, social value, health status or emotional stability. In one of his papers, the sociologist Rosa proposes the following definition of risk: "*a situation or event where something of human value (including humans themselves) has been put at stake and where the outcome is uncertain*" [44].

Three main characteristics of the concept of risk can be extracted from this definition that apply in general, regardless of the context or the type of risk you are going to consider. First,

risk relates to a possible situation in which someone's interests, whether the individual person or the entire organization, are affected in some way; either positively or negatively. Second, risk is closely related to the concept of *probability* and *uncertainty*. We cannot speak of risk when we consider events that will occur with absolute certainty, but only when we speak of events that could, with some degree of uncertainty, occur sooner or later in the future. Finally, the concept of risk is related to the concept of possibility, so one cannot speak of risk for impossible events.

In cybersecurity the following definition of risk can be adopted form NIST: "*An effect of uncertainty on or within information and technology. Cybersecurity risks relate to the loss of confidentiality, integrity, or availability of information, data, or information (or control) systems and reflect the potential adverse impacts to organizational operations (i.e., mission, functions, image, or reputation) and assets, individuals, other organizations, and the Nation*" [45].

Cyber risk can be identified, and usually assessed, as the result of three competing factors: *threat*, *vulnerability* and *impact*. The analysis of these three basic components enables the identification of strategies to be adopted for risk reduction and mitigation. These may be reducing the vulnerabilities present in the system, adopting mechanisms and strategies for impact mitigation, or addressing the reduction of the considered threat. Each organization must assess its own risks in accordance with its level of risk tolerance and choose the appropriate countermeasures. There is always some residual risk to be considered because, in general, it is thought to be not possible to minimize a risk to zero due to the concept's close relationship to the random character of the variables that create it. Many strategies, including *risk avoidance*, *risk transfer*, *risk mitigation*, and *risk acceptance*, can be used to lower hazards. Risk avoidance goal is to eliminate the risk by avoiding the source that introduced the risk. Risk transfer goal is to transfer the risk to another party, usually through an insurance with an insurance company. Risk mitigation goal is to implement and introduce the necessary technologies, policies and process to lower the level of risk, for example by reducing the likelihood of an adversarial event happening or reducing its impact on the organization. Risk acceptance, instead, involves accepting the risk, usually when it is under a certain threshold, and adopt the necessary measures to manage and control it.

The risk associated with the occurrence of cyber attacks that could endanger the security of information or operations must be managed by all businesses, whether they are public or private and engaged in a wide range of productive sectors. Recently, a variety of frameworks have been developed on a global level to guide the implementation of security measures to prevent and minimize risk using the appropriate governance structure and mechanism. Some of these

frameworks are generalist in nature and are therefore adaptable to any context, while others have been designed and developed within the framework of specific manufacturing sectors, capturing their specific characteristics. Some are free to adopt (e.g., NIST Cybersecurity Framework [10]), while others allow one to embark on an independent auditing journey that can result in true certification (e.g., ISO 27000 [46]). In general, a cybersecurity framework provides adopters with a methodology for analyzing the current level of risk, setting a goal to achieve, and identifying the things needed to achieve it.

Unavoidably, an organization will need to evaluate how successfully the security measures presently in place allow it to achieve the objectives specified by the end goal, regardless of the type of cybersecurity framework it chooses. This practice is known as **Cybersecurity Assessment**, and it allows for periodic evaluation of progress in implementing a program aimed at increasing the level of cybersecurity. In the various proposed methodologies, a key role is played by risk evaluation, a process known as **Risk Assessment**. This process, as introduced earlier, allows the risk deriving from cyber threats to be assessed so that the resilience of the system against them can be evaluated and, based on a cost-benefit analysis, activities to better contain the risk and improve the security posture can be planned. Several methodologies, with different levels of detail, have been proposed in this area over time and in general, two macro-approaches can be distinguished:

- **Qualitative Assessment**: methodologies attributable to this approach are based on the assumption that the practices and solutions being evaluated do not allow objective quantification of their quality. They therefore prefer to go for subjective elements and aspects that are difficult to measure.
- **Quantitative Assessment**: this approach assumes that what is being evaluated is measurable through a set of appropriately defined metrics. This measurement should be carried out as objectively as possible, seeking to limit any contamination of the measured data resulting from subjective considerations.

When a "leaner" approach to assessment is desired, or when assessment processes must be completed quickly or inexpensively, qualitative assessment approaches are frequently used. On the other hand, quantitative approaches are more suitable when it comes to periodic monitoring of the effectiveness of a cybersecurity governance process or when it comes to calculating the return on investment for a cybersecurity investment. The two methods can therefore be used together rather than separately to undertake a comprehensive cybersecurity evaluation.

3.1.1 NIST Cybersecurity Framework

The U.S. National Institute of Standards and Technology (NIST) developed the first version of the cybersecurity framework (namely, Framework for Improving Critical Infrastructure Cybersecurity) in 2014 as part of its mission to promote U.S. innovation and industrial competitiveness. The document has been revised over the years and a new, updated, version has been released in 2018 [47], which is currently the reference document. The NIST Framework has been designed and developed targeting especially critical infrastructure in order to provide a set of mechanisms that can be adopted to improve their overall cybersecurity posture, but this does not prevent to adopt the NIST Framework by organizations operating in any sector.

The NIST Framework aims to analyze and leverage an organization's business objectives to define and guide cybersecurity operations in a way that considers and integrates cybersecurity risk assessment and management as part of the organization's own risk management process. The NIST Framework consists of three parts: (ii) *Framework Core*, (ii) *Implementation Tiers*, and (iii) *Framework Profiles*. The Framework Core is a set of cybersecurity initiatives, findings, and learning materials that all critical infrastructure and commercial sectors may use and share. The Core's components provide thorough instructions for developing distinctive *Profiles* for each organization based on the application context. By leveraging *Profiles*, the Framework will help an organization prioritize its cybersecurity activities in line with its business/mission needs, risk tolerance, and resource availability. Companies may utilize the specified *Implementation Tiers* as a set of principles and methods to examine and grasp the organization's strategy and created processes for cybersecurity risk management to prioritize and achieve the intended cybersecurity goals.

In order to provide a common organizational structure for multiple approaches to cybersecurity, the NIST Framework pulls together standards, regulations, and practices that have previously proven to be effective. Because it makes use of globally recognized cybersecurity standards, the NIST Framework may also be used as a model for international collaboration on increasing cybersecurity in critical infrastructure as well as other sectors and communities. It offers an adaptable framework for managing cybersecurity, taking into account how it affects the human, computer, and physical components. It can be adopted regardless of whether a company's primary cybersecurity focus is on information technology (IT), industrial control systems (ICS), cyber-physical systems (CPS), or connected devices more generally, such as the Internet of Things (IoT). The NIST Framework provides a common means for organizations to:

- Describe and evaluate their current cybersecurity posture.
- Identify which are the goals upon improving their cybersecurity posture.

- Evaluate progress towards the identified goals.
- Share results of the evaluation and cybersecurity risk to internal and external stakeholders.

Framework Core

The Core provides those adopting the methodology described in the framework with a set of activities to be undertaken to achieve certain levels of security. These do not represent a checklist of actions to be taken, but rather a set of cybersecurity concepts that are useful in managing cybersecurity risk. The four elements that make up the Core are:

- **Functions**: at the highest level, functions categorize and group fundamental cybersecurity activities which are identified as: *Identify*, *Protect*, *Detect*, *Respond*, and *Recover*. By organizing information, facilitating risk management choices, mitigating threats, and improving by drawing lessons from past actions, they help a company communicate its management of cybersecurity risk. The Functions also support current incident management approaches and aid in demonstrating how investments in cybersecurity are paying off. Investments in planning and exercises to foster prompt reaction and recovery activities, resulting in less of an impact on service delivery.
- **Categories**: each Function is divided into Categories which further details cybersecurity activities and processes to implement which are required to correctly support the given Function.
- **Subcategories**: Categories are further divided into Subcategories which go further into the detail of the technical activities or management processes to implement. These provide a set of concrete results that are expected from the correct application of a given Category to better guide an organization in the implementation of security policies. Examples of Subcategories include “External information systems are cataloged”, “Data-at-rest is protected”, and “Notifications from detection systems are investigated”.
- **Informative References**: consists in specific parts of industry-wide norms, policies, and procedures that provide examples of how to accomplish each Subcategory’s objectives. These are mostly illustrative, however they serve as references to the industry’s most prevalent standards.

The five Functions identified in the NIST Framework describe activities that should be adopted by the organization as a continuous process, rather than performed once to achieve a desired state; they are taught to provide guidelines to the organization in developing a structured approach to dynamic cybersecurity risk management. The Functions are:

- **Identify**: is a collection of those activities aiming to create organizational awareness of cybersecurity risk to systems, people, assets, and data. For the Framework to be used effectively, the Identify Function's activities are fundamental. An organization may concentrate and prioritize its efforts in accordance with its risk management strategy and business requirements by having a thorough understanding of the business environment, the resources supporting important operations, and the associated cybersecurity threats. *Asset Management*, *Business Environment*, *Governance*, *Risk Assessment*, and *Risk Management Strategy* are a few examples of Categories within this function.

- **Protect**: is a collection of those activities aiming to create and execute suitable measures to guarantee the delivery of the organization essential services. The Protect Function consider those activities that provide the capability of limiting or containing the effects of a possible adverse cybersecurity event. Examples Categories under this function include *Identity Management* and *Access Control*, *Awareness and Training*, *Data Security*, *Information Protection Processes and Procedures*, *Maintenance*, and *Protective Technology*.

- **Detect**: is a collection of those activities aiming to create and put into operation the necessary processes and technologies to promptly and quickly detect cybersecurity incidents. Categories within this Function include: *Anomalies and Events*, *Security Continuous Monitoring*, and *Detection Processes*.

- **Respond**: is a collection of those activities aiming to create and implement the necessary actions to respond to an identified cybersecurity issue. The Respond Function provides the organization guidelines to improve the capacity to reduce the undesirable outcomes of a possible cybersecurity event. Categories within this Function include: *Response Planning*, *Communications*, *Analysis*, *Mitigation*, and *Improvements*.

- **Recover**: is a collection of those activities aiming to create and put into operation the necessary actions to maintain resilience plans and to restore any capabilities or services that were harmed as a result of a cybersecurity event. The Recover Function encourages prompt return to routine business activities in order to lessen the effect of a cybersecurity event. Categories within this Function include: *Recovery Planning*, *Improvements*, and *Communications*.

Implementation Tiers

The NIST Framework Implementation Tiers provide insight into how the organization views cybersecurity risk and the processes it employs to manage this risk. The Tiers range from

a low value, Partial (Tier 1), to the highest value, Adaptive (Tier 4), and are used to describe the complexity and rigor of cybersecurity risk management processes and practices in an increasing way; both those currently adopted by the organization and the level to which it wants to aspire. They also allow for an assessment of how well cybersecurity risk management processes are integrated within the organization and how well these are considered and cooperate in overall risk management.

Tier selection takes several aspects into consideration, from current risk management practices to the threat environment, while also considering the organization's regulatory requirements, business objectives, supply chain, and so on. When an organization is selecting the target Tier to aspire to, it must consider what its needs are in terms of the feasibility of implementing the selected Tier and how the selected Tier can reduce the cybersecurity risk level to assets and resources to a level that is acceptable. Implementation Tiers are the following:

- Tier 1 - Partial: The management of cybersecurity risks in the organization is informal and not well-structured. The organization's approach to cybersecurity risk management is reactive and lacks a clear understanding of the organization's objectives, threat environment, and business requirements. The organization has limited knowledge of cybersecurity risks and has an inconsistent approach to managing them due to differences in experience and external sources of information. There is a lack of processes for sharing cybersecurity information within the organization. Additionally, the organization does not understand its role in the larger ecosystem and does not collaborate with other entities to share threat intelligence, best practices, or technologies. Finally, the organization is unaware of the cyber supply chain risks associated with the products and services it uses and provides.

- Tier 2 - Informed Risk: The organization has approved risk management practices, but they may not be formalized as an organization-wide policy. The organization's prioritization of cybersecurity activities is based on its risk objectives, threat environment, and business requirements. While there is some awareness of cybersecurity risks, the organization lacks an established approach to managing these risks across the entire organization. Cybersecurity information is shared informally within the organization, and consideration of cybersecurity occurs at some but not all levels of the organization. Cyber risk assessments occur, but they are not regularly repeated. The organization understands its role in the larger ecosystem with respect to either its dependencies or dependents, but not both. While the organization collaborates with other entities and generates some of its own information, it does not consistently share this information with others. Finally,

the organization is aware of the cyber supply chain risks associated with the products and services it uses and provides, but it does not consistently or formally address these risks.

- Tier 3 - Repeatable: The organization's risk management practices are approved and expressed as policy. They are regularly updated based on changes in business/mission requirements, the threat and technology landscape, and risk management processes. The organization has an established approach to managing cybersecurity risks, with policies, processes, and procedures in place to define, implement, and review risk-informed decisions. Consistent methods are used to respond to changes in risk, and personnel have the necessary knowledge and skills to perform their roles. Cybersecurity risks are consistently and accurately monitored for organizational assets, and senior executives communicate regularly about cybersecurity risk. Consideration of cybersecurity occurs throughout the organization's operations, and it understands its role, dependencies, and dependents in the larger ecosystem. The organization collaborates with other entities, both receiving and sharing information. It is aware of cyber supply chain risks associated with the products and services it provides and uses, and it usually acts formally upon those risks with mechanisms such as written agreements and governance structures.
- Tier 4 - Adaptive: The organization continually improves its cybersecurity practices by learning from past and current activities and using advanced technologies to respond effectively to evolving threats. There is an organization-wide approach to managing cybersecurity risk with policies and procedures in place to address potential events. Senior executives monitor cybersecurity risk and consider it when making decisions. The budget is based on risk tolerance, and business units analyze risks in the context of the organization's risk tolerance. Cybersecurity is part of the organizational culture, and the organization quickly adapts to changes in business objectives. The organization understands its role in the larger ecosystem and shares information with collaborators. It uses real-time information to understand cyber supply chain risks and communicates proactively with suppliers.

Framework Profiles

The Profile describes how the Functions, Categories, and Subcategories of cybersecurity align with the organization's business requirements, risk tolerance, and resources. It enables organizations to create a roadmap for reducing cybersecurity risk that is aligned with their goals, legal/regulatory requirements, and risk management priorities. Large organizations may have multiple profiles to suit different components and their individual needs.

Framework Profiles can describe the current or desired state of cybersecurity activities. The Current Profile shows the current outcomes achieved, while the Target Profile shows the outcomes needed to achieve desired cybersecurity risk management goals. Profiles are useful for business/mission requirements and for communicating risk within and between organizations. There are no prescribed Profile templates, giving flexibility in implementation.

Comparing Profiles (e.g., Current Profile vs. Target Profile) can identify gaps to address cybersecurity risk management objectives. An action plan can be created to mitigate these gaps to achieve a given Category or Subcategory, contributing to the roadmap mentioned above. Prioritizing the gaps to address is based on the organization's business needs and risk management processes. This risk-based approach helps organizations gauge the resources needed (e.g., staffing, funding) to achieve cybersecurity goals in a cost-effective, prioritized manner. The Framework is a risk-based approach, and the applicability and fulfillment of a given Subcategory are subject to the Profile's scope.

Framework Application

The following steps illustrate how an organization could use the Framework to create a new cybersecurity program or improve an existing program. These steps should be repeated as necessary to continuously improve cybersecurity. The six fundamental steps identified in the NIST Framework are:

- Step 1 - Prioritize and Scope: the organization delineates its business/mission objectives and overarching organizational priorities. Based on this data, the organization strategically decides on cybersecurity implementations and ascertains the range of systems and assets that underpin the chosen business line or process.
- Step 2 - Orient: After defining the scope of the cybersecurity program for a given business line or process, the organization proceeds to identify relevant systems and assets, regulatory requirements, and overall risk management approach. To identify potential threats and vulnerabilities applicable to the identified systems and assets, the organization may consult various sources.
- Step 3 - Create a Current Profile: the organization creates a Current Profile by specifying the Framework Core Category and Subcategory outcomes that are presently being attained. This step helps establish a baseline for subsequent actions and facilitates the process by recording partially achieved outcomes.
- Step 4 - Conduct a Risk Assessment: the evaluation process can be informed by the organization's existing risk management procedures or prior risk assessment endeavors.

Through a thorough analysis of the operational landscape, the organization can determine the probability of a cybersecurity incident and assess its potential impact. In order to effectively anticipate and mitigate threats, it is crucial for organizations to identify emerging risks and utilize cyber threat intelligence from both internal and external sources.

- Step 5 - Create a Target Profile: the organization formulates a Target Profile that evaluates the Framework's Categories and Subcategories, elucidating the preferred cybersecurity outcomes. Additionally, organizations may create their own unique Categories and Subcategories to address organizational risks. When creating the Target Profile, the organization should take into account external stakeholders, such as sector entities, customers, and business partners. Moreover, the Target Profile should align with criteria within the target Implementation Tier.

- Step 6 - Determine, Analyze, and Prioritize Gaps: the organization conducts a gap analysis by comparing the Current Profile and the Target Profile to identify areas that need improvement. Based on the analysis, the organization creates a prioritized action plan that reflects the mission objectives, cost-benefit analysis, and risk assessment to achieve the outcomes defined in the Target Profile. The organization should also evaluate the required resources, such as funding and workforce, needed to address the gaps. By utilizing Profiles in this manner, the organization can make informed decisions regarding cybersecurity activities, promote risk management, and facilitate cost-effective and targeted improvements.

- Step 7 - Implement Action Plan: After identifying gaps between the Current Profile and the Target Profile, the organization decides which actions to undertake to bridge the gaps and achieve the desired cybersecurity outcomes. The Framework provides Informative References for the Categories and Subcategories as examples, but organizations should determine which sector-specific standards, guidelines, and practices suit their requirements best. The organization should adjust its existing cybersecurity practices accordingly to align with the Target Profile, thereby fostering informed decision-making, promoting risk management, and ensuring cost-effective and targeted improvements.

An organization may repeat the steps outlined above as necessary to continually evaluate and enhance its cybersecurity posture. For instance, organizations might discover that more frequent repetition of the orientation phase improves the quality of risk assessments. Additionally, organizations could track progress through successive updates to the Current Profile, followed by a comparison of the Current Profile with the Target Profile. Organizations can also use

this process to align their cybersecurity program with the desired Implementation Tier of the Framework.

Risk Assessment Methodology

NIST in the series of documents published for supporting information security released also a document describing and detailing a methodology for cybersecurity risk assessment. Special Publication 800-30 [1] has the goal to provide guidelines for conducting risk assessment of IT systems; the target are critical infrastructures, but the methodology can be freely and effectively applied in different contexts concerning information technology. The publication details all the steps in the risk assessment process and how the results of the infrastructural and operational analysis can be used to identify the correct actions to mitigate and respond to the identified risk; thus going on to integrate these into the overall enterprise risk management process. The main steps identified in the risk assessment process are (i) the preparatory phase of the assessment (e.g., identification of assets, identification of business goals, and so on); (ii) the phase of actual risk analysis and how this can be conducted effectively; (iii) the phase of communicating the results of the assessment to stakeholders; and (iv) the phase of maintaining the assessment as a continuous and cyclical process over time through which the enterprise can rapidly and promptly adapt to changes in risk factor exceeding the acceptable level.

In the NIST SP 800-30 special emphasis is placed on the fact that the risk assessment process is not an activity to be carried out only once, but that it needs to be embraced and implemented by organizations as a cyclical process that also serves to be able to account for variations and continuous changes both at the level of business assets and threats from cyberspace; the risk assessment activity should be integrated within the system development life cycle.

The methodology proposed by NIST, although it goes to consider as an objective the risk analysis of information systems takes into account, both in the risk models proposed, and in the activities to be carried out and things to be considered, events that may go to compromise the proper functioning of the system or that may cause some kind of damage in general; therefore considering both events of a computer type (i.e., cyber attacks), but also events of a natural type (i.e., fires, floods, and so on).

Risk assessments discuss the potential negative effects of the operation and use of information systems and the information processed, stored, and transmitted by those systems on organizational operations and assets, on individuals, on other organizations, and on the economic and national security interests of the United States. In order to identify risks that are

similar to the organization's primary missions/business functions, mission/business processes, mission/business segments, common infrastructure/support services, or information systems, organizations undertake risk assessments. The proposed risk methodology, aims to provide information that can be used in the risk management process to define, among other things:

- The development of an information security architecture.
- The design and implementation of security solutions for IT systems that takes into account technology to adopt, policies to enforce and selection of supply chain.
- The implementation of continuous monitoring strategies for both risk and security solutions.

Cybersecurity risk can be viewed as the level to which a system is exposed to potential threats should they come from within or without. Risk can be assessed by going to analyze several factors: (i) the impact that the threat, should it materialize, has on the system and things connected to it; (ii) the likelihood with which the threat might occur Cybersecurity risks are those risks that have effect on the confidentiality, integrity, or availability of systems and information and reflect the possible impact to the organization (i.e., business goal, services, image and reputation), assets and everything that is interconnected and in relation to the system and the organization.

The risk assessment methodology proposed by NIST includes: (i) a process to evaluate the risk; (ii) a reference risk model, which define relevant risk factors in the context of application; (iii) how the assessment is conducted, in other word, which are the values associated to risk factors and how these values can be combined to evaluate the risk level; (iv) an analysis approach, which define the point of view of the assessment (i.e., threat-oriented, asset-oriented, vulnerability-oriented) and what are the key points to consider in order to provide an adequate coverage of the problem space.

In the NIST SP 800-30 it is provided a reference risk model detailing which are the fundamental risk factors to consider in the assessment and the relationships among those factors. Risk factors serve as input during the assessment to analyze the system posture and determine the level of risk. These are identified as *threat*, *vulnerability*, *impact*, *likelihood*, and *predisposing condition*. Through the analysis and evaluation of these risk factors with respect to the organization under consideration and possible then assess the level of risk, understand what are the major issues and identify the best strategies to mitigate or eliminate the various risk factors to bring the level of risk to an acceptable level.

Threat

In the NIST SP 800-30 a threat is defined as "any circumstance or event with the potential to adversely impact organizational operations and assets, individuals, other organizations, or the Nation through an information system via unauthorized access, destruction, disclosure, or modification of information, and/or denial of service". Threat sources can be identified as:

- hostile attacks, also referred as *adversarial threats*;
- human errors of omission or commission;
- structural failures of organization resources and assets;
- failure beyond the control of the organization, including natural and man-made disasters.

While each of the threat sources can be characterized as *intentional* or *unintentional*. Thus going to represent both the fact that a malicious event may arise from the intention, whatever the motive, of an attacker to cause harm to the organization, and the fact that a malicious event may arise from some kind of unintentional error or forgetfulness.

Vulnerabilities and Predisposing Conditions

NIST defines a vulnerability as "a weakness in an information system, system security procedures, internal controls, or implementation that could be exploited by a threat source." Most vulnerabilities can be identified as the lack (whether intentionally or unintentionally inserted) of security control measures, or they can be identified as the presence of security control measures that nonetheless contain within them weaknesses that can be exploited to disable or bypass them. Obviously, this consideration should not be viewed as an assessment with respect to an exact instant in time, but one must keep in mind that vulnerabilities can evolve and increase over time, new vulnerabilities can be discovered (and are discovered every day) or introduced as the organization's infrastructure evolves.

NIST defines a predisposing condition as "a condition that exists within an organization, a mission or business process, enterprise architecture, information system, or environment of operation, which affects (i.e., increases or decreases) the likelihood that threat events, once initiated, result in adverse impacts to organizational operations and assets, individuals, other organizations, or the Nation". Predisposing conditions can be the fact that a system has no network connection to the outside world, this, for example, lowers the likelihood that a system will be attacked via the internet, while a system or factory that is placed in close proximity to areas with a high risk of flooding or fire goes to increase the possibility of these events occurring.

For example, a lack of contingency preparations, the use of antiquated technology, or weaknesses/deficiencies in information system backup and failover processes might all result in vulnerabilities that cannot be quickly fixed. These vulnerabilities invariably increase the likelihood of threat occurrences having negative effects on businesses. The total security posture of corporate information systems and operating environments includes vulnerabilities, including those attributable to predisposing factors, which might have an impact on the likelihood of the occurrence of threat events.

Impact

NIST defines the impact as "the magnitude of harm that can be expected to result from the consequences of unauthorized disclosure of information, unauthorized modification of information, unauthorized destruction of information, or loss of information or information system availability'. Thus, in risk assessment, it is important for an optimal outcome, to assess what the consequences of a threat event may be so that we can identify, the events that may do the most harm, to be able to understand whether a risk can be considered acceptable or not and possibly assess the trade-of between the cost to mitigate a risk and the impact it would have.

Likelihood

NIST define Likelihood of occurrence as "a weighted risk factor based on an analysis of the probability that a given threat is capable of exploiting a given vulnerability (or set of vulnerabilities)";in other words, likelihood expresses the probability that a certain event will occur and the probability that it will cause some damage to the system under consideration.

For adversarial threats, likelihood assessment typically takes into account: (i) adversary intent, which are the goals of attackers; (ii) adversary capability, which are or how many resources attackers have at disposal for the attack; and (iii) adversary targeting, whether or not the exact purpose of attackers is to cause some damage to the organization. For other types of threats (non-adversarial), likelihood assessment should be done by referring to historical data about occurrence or empirical data. Other factors that can be used when assessing likelihood of occurrence can be the presence or absence of security measures, predisposing conditions and so on.

In Figure 3.1 is reported the general risk model proposed by NIST.

Conducting Risk Assessment

The first step in the assessment process is the preparation phase for the assessment. In this case, the goal is to: (i) identify and delineate the application context in which the assessment will take place, the methodology to be adopted, and so on; (ii) identify and define the purpose

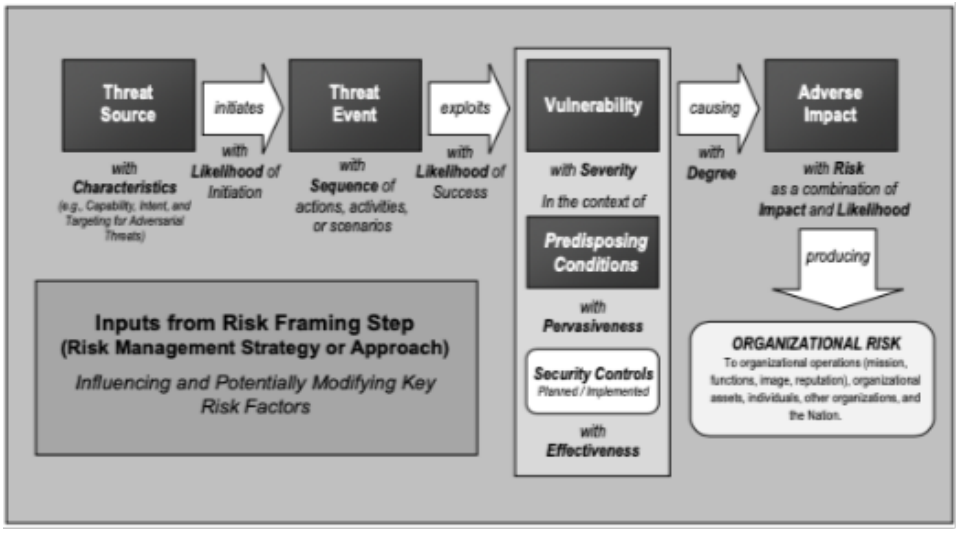


Fig. 3.1 NIST Proposed Risk Model [1]

of the 'assessment, that is, what information the assessment is to produce and what decisions the results of the assessment are to support; (iii) identify what data are to be used to perform the assessment and what sources are to be used such as threats, vulnerabilities, and impact; (iv) identify the risk model to be used and define the methodology to be adopted for risk analysis and assessment.

The second step is the actual execution of the risk assessment activity. Here all the factors that make up a risk factor for the organization must be assessed. The various levels of risk must be ordered from highest to lowest so that they can be prioritized according to which decisions to mitigate or eliminate the risk must be executed. To do this, the organization must analyze the possible threats, the vulnerabilities present, and the impact the threats may have as well as the likelihood (likelihood) that they may occur. Performing the risk assessment must include the following activities:

- Identify the threat sources that are relevant to the organization, evaluating their various characteristics such as the possibilities of attackers.
- Identify the events that result from the identified threat sources.
- Identify and assess the vulnerabilities in the organization that can be exploited to initiate a threat event, assess their predisposing conditions and probability factors that can lead to a certain type of impact on the organization.
- Determine the likelihood that a certain threat event may occur and the likelihood that it may lead to a certain type of impact on the organization.

- Determine and quantify the impact that a threat event may have on the organization and its assets, individuals, and anything else related to the organization (such as external systems and organizations)
- Finally, calculate the actual information security risk as a composite of all its component factors, as a combination of the likelihood that a certain threat event will materialize and the impact it may have.

The fourth and final step in the risk assessment process is to define the activities and processes necessary to maintain the assessment as an ongoing activity over time. The output of the risk assessment process must serve to support risk management processes and in providing data to make informed decisions with respect to which aspects should be prioritized or which and what type of risk mitigations should be adopted. It is therefore necessary for the risk assessment process to be continuous in order to adapt to possible changes that may shift and alter the outcome of the assessment.

So, to recap, NIST provides the following formula for calculating risk:

$$Risk = Likelihood \times Impact \tag{3.1}$$

A number of metrics by which risk can be assessed are also provided. These consider and evaluate risk factors according to levels, from low to high, which can be combined through a matrix to calculate the overall level of risk.

3.1.2 Italian Cybersecurity Framework

The Italian Cyber Security Framework [48] represents a tool to identify possible deficits within the cybersecurity management of an organization, both in the public and in the private sector and to define a risk management strategy that persists as the threat and technology change.

The Italian Framework is based on the "Framework for Improving Critical Infrastructure Cybersecurity" [47], developed by the US National Institute for Standards and Technology (NIST), and extended and characterized for the Italian enterprise ecosystem. The NIST Framework has been designed and developed with critical infrastructure as a target, hence it may introduce a complexity level that may not be appropriate in the context of Italian companies. The Italian business landscape is mostly composed of small and medium-sized enterprises and this certainly poses challenges in the drafting first and implementation later of the Framework. It must be taken into consideration that small and medium-sized enterprises usually do not

possess the skills internally to manage security and nor, usually, do they possess the economic budget to do so.

Moreover, in 2018, at the European level, the General Data Protection Regulation (GDPR) [49], which governs the processing and circulation of personal data, became operational. The Regulation imposed a change of perspective concerning the protection of personal data by introducing the principle of *accountability*, and its implementation is a key step for any organization that intends to process personal data.

As a result of this, in 2019, the national cyber security framework was updated by introducing the concepts and controls related to the processing of personal data regarding what is stated in the GDPR [43]. In particular, elements have been integrated from the previous version of the framework to take into account security issues introduced by the supply chain and authentication and identity management processes. In addition, for compliance with the GDPR, elements have been introduced to address the security and proper management of personal data in case of cybersecurity incidents.

In particular, the following aspects have been added to address the issue of data protection:

- Data management processes, with particular reference to those applicable to personal data.
- The way personal data are processed.
- Roles and responsibilities in personal data management.
- The data protection impact assessment.
- The methods of documentation and communication following incidents that amount to a personal data breach.

The Italian Cybersecurity Framework derives three fundamental concepts from the NIST Framework: **Framework Core**, **Profile** and **Implementation Tier**.

Framework Core

The core represents the life cycle structure of the management process of cybersecurity, both from a technical and organizational point of view. The core is structured hierarchically into *Function*, *Category* and *Subcategory*. Concurrent and continuous Functions are: *Identify*, *Protect*, *Detect*, *Respond*, *Recover*, and they represent the main topics to deal with in order to strategically obtain an appropriate cyber risk management.

The 5 Functions are inherited from the NIST Cybersecurity Framework [47]. The Italian Cybersecurity Framework integrates the Function with the obligations defined within the GDPR. A new Category and nine new Subcategories have been included. These go into more detail about: data management processes, how personal data is handled and roles in the management of personal data, personal data impact assessment, and how to report and document cyber incidents that result in a personal data breach.

Profiles

Profiles represent the set of Subcategories selected by an organization. Subcategories are selected based on the applicability they have with respect to the organization and several other factors such as the outcome of the risk assessment, business context and so on. Profiles can be used both to describe the organization's current security posture (*Current Profile*), but also to describe the outcome they want to target in terms of security posture (*Target Profile*). The comparison between the *Current Profile* and *Target Profile* is used to identify those areas that need an improvement and which activities should be taken to reach the desired goal. It is envisaged that the Subcategories may be supplemented with additional practices not included in the Framework in order to comprehensively manage risk. The progress made toward the Target Profile can then be prioritized using the Current Profile. Profiles can also be used as a means of performing a self-assessment on the organization's security posture and as a means of communicating, sharing and comparing the results obtained from the security posture assessment in general.

Implementation Tier

Implementation Tiers are used as a means of assessing the Cybersecurity Risk Management process adopted by the organization and how well it is integrated into business processes. Four tiers of assessment are provided, from weakest to strongest, and are inherited by the one presented in [47]: *Partial*, *Informed*, *Repeatable*, and *Adaptive*.

Moreover, identified within the Framework are priority levels and maturity levels that organizations adopting the framework can use to understand the actions needed, for proper risk management, to arrive at the identified target profile.

Priority Levels

With the help of the priority levels, organizations and businesses may define a program of actions that will most significantly lower the levels of risk they are now exposed to. The overarching objective is to determine the critical activities (Subcategories) that must be taken right away in order to support risk analysis and management operations. The determination of priority levels assigned to subcategories should be conducted based on two specific criteria:

- The ability to lower cyber risk by addressing one or more of the following factors: (i) threats exposure, which is the set of factors that affect how likely it is that a threat will materialize; (ii) likelihood of threats occurrence, which is how likely it is that a threat will occur over time; and (iii) impact on business (assets and goals), which is the degree of harm brought on by the threat's occurrence.
- The ease of use of Subcategories, taking into account the average degree of organizational and technical development needed to accomplish the particular action.

Maturity Levels

With Maturity Levels it is possible to quantify a security process' maturity, the maturity of how a certain technology is being implemented, or the quantity of suitable resources that have been used to implement a certain Subcategory. Each organization may use maturity levels as an indicator to assess how well it has implemented Subcategories and to define objectives and priorities for progress. Subcategory maturity levels should be identified according to the following characteristics:

- Specificity by subcategory: it is possible to have different maturity levels for different subcategories in an organization.
- Completeness of security practices: the maturity level of a Subcategory is identified by taking into consideration the level of implementation of all security practices related to that Subcateogry.

Maturity levels are expressed according to the Capability Maturity Model Integration (CMMI) scale [50], which has five levels:

- Initial: Control implementation relies on processes, procedures, and technical solutions with results that are unpredictable, undocumented, unorganized, and often executed ad-hoc. Successful management relies on the individual skills of staff and not on the proven use of well-defined processes.
- Repeatable: Control implementation makes use of well-defined and documented processes, procedures and technical solutions in each or a subset of the organization's functions involved, but in a manner that is not consistent at the level of corporate regulations (each function manages its own processes, procedures and technical solutions independently).

- Defined: The implementation of control uses well-defined, documented and standardized processes, procedures and technical solutions at the level of corporate regulations. The various functions can specialize their own processes, starting with those standardized at the level of corporate regulations.
- Managed: In addition to including aspects of the "Defined" maturity level, quantitative targets are set for the performance of the processes, procedures and technical solutions underlying the implementation of control. The effectiveness of processes, procedures and technical solutions is monitored and measured quantitatively.
- Optimized: In addition to including aspects of the "Managed" maturity level, the processes, procedures, and technical solutions underlying control implementation undergo continuous improvement in response to changes in the organization and considering past experiences.

Framework Application

The application of the Framework follows the same guidelines as defined in the NIST Framework [47]. The goal is to provide a set of operations to be performed systematically to guide organizations in using the Framework. It all starts with contextualizing the Framework, which involves identifying the regulatory context in which the organization operates and then identifying what are the categories and subcategories that according to the relevant regulatory framework are essential. Then the organization proceeds to identify its business goals, systems, and assets that are critical to the organization. With this information and by analyzing the cybersecurity processes and technologies adopted, the organization identifies the Current Profile and performs the Risk Assessment process (see Section 3.1.2). At this point, the organization can select the Target Profile, identify what the current deficiencies are, and plan the activities necessary to achieve the set goal.

Risk Assessment Methodology

In 2021, the cybersecurity assessment methodology was drafted based on the Italian Cybersecurity Framework [51], which is the set of operations that organizations can follow and implement to measure their posture in terms of cybersecurity.

The methodology identifies three main steps concerning risk assessment:

- **Contextualization**: This first phase aims to contextualize the Framework to the reality of interest. The product of this phase will be a Target Profile, that is, the objective in terms

of security posture to be achieved, to which to strive and on which the assessment is carried out. Its proper definition is then functional to the conduct of the next two phases.

- **Measure**: In this second phase, the current cybersecurity posture of the organization is identified against what is defined in the Target Profile. This process is done through interviews with individuals who are competent for the specific analysis needs. The result of the interviews is expressed in terms of coverage and maturity for each identified control.

- **Evaluation**: In the third phase, the results of the measurement phase are evaluated. This operation makes it possible to calculate, from the coverage and maturity values of each subcategory identified, the level of deviation of the organization from the target profile compared to the current profile. These assessments can later be used to analyze the results of the assessment process and identify the activities needed to achieve the target profile.

Contextualization

The objective of this first phase is to contextualize the Framework with respect to the existing case and define for the organization or enterprise (at least) a *Target Profile*, which represents the desired cybersecurity posture, the level to be achieved. This phase therefore defines the perimeter within which the organization intends to move (the contextualization), in terms of cybersecurity posture, and the goal that the organization aims to achieve (the Target Profile), which will then be compared, in the assessment phase, with the organization's current posture (captured in the measurement phase).

Contextualization consists of identifying and selecting the subcategories of the Italian Cybersecurity Framework that are of interest and relevant to the case study. This selection depends on many factors, both related specifically to the organization and cross-cutting to individual realities, such as, for example, industry and/or general regulations, norms, and standards.

Thus, the creation of contextualization is done through a process involving the following steps:

- Subcategory selection: The selection process is performed by referring to identified *prototypes*. The prototypes describe within them which subcategories are mandatory, which are recommended, and which are free; that is, those left to the discretion of the organization. All subcategories that are "mandatory" must necessarily be selected if

a contextualization consistent with the combination of the selected prototypes is to be created.

- Definition of priority levels: at this stage, priority levels are defined by considering (i) the priority levels possibly assigned by the prototypes themselves; (ii) context information from the specific reality creating the contextualization. Thus, by appropriately combining factors (i) and (ii), an initial assignment of priority levels must be made, which may be supplemented and revised after the measurement phase.
- Application guidelines: application of the guidelines contained in the accompanying prototype documents. These guidelines can be used for the target profile identification phase and for the definition of controls.
- Definition of maturity levels: possible definition of maturity levels for each selected subcategory. The description of the maturity levels for each subcategory defines the criteria to be met to achieve the different maturity levels expected.

The target profile represents the profile against which the assessment results are calculated. In fact, the assessment phase performs a deviation analysis between the current profile identified in the measurement phase and one or more target profiles defined in the contextualization phase. A target profile defines a set of security controls that must be implemented to achieve the desired cybersecurity posture by mapping them onto the subcategories of the Framework.

Measure

The measurement step is carried out according to the target profile identified in the contextualization. Starting from this, the assessment is carried out in 4 steps, the first 3 being mandatory and the last one optional:

- Step 1: Creation of the questionnaire.
- Step 2: Administration of the questionnaire.
- Step 3: Consolidating the questionnaire and obtaining the current profile.
- Step 4: Remodeling the priority levels (optional).

In the questionnaire creation phase, starting with the target profile and for each of the resulting checks to be made, the following fields are defined:

- *Response*: free text field where to report the information collected during the administration of the questionnaire itself.

- *Notes about degree of coverage*: field necessary to report, in text form, information about the coverage status of the control.
- *Degree of coverage*: field needed to report, in numerical form, the degree of coverage of the control (how well the control is met). The degree of coverage is expressed with a qualitative scale from 0 to 1. Where 0 indicates no implementation of the control and 1 indicates total implementation.
- *Notes about maturity level*: field in which to report, in text form, details about the maturity status with which a control or part of a control is implemented.
- *Maturity level*: field required to report the maturity level for the entire control. The enhancement of this field is based on the five levels defined by CMMI. In cases where controls have zero coverage, the maturity level should be reported as "Not Applicable" (N/A).
- *Evidence*: evidence supporting the estimated coverage and maturity level will be reported in this field.
- *Notes*: field in which to report any additional notes that emerged during the assessment phase.

After the questionnaire is created, it is administered to the organization with which the assessment is being conducted.

Once the results of the questionnaire administration are collected, they are analyzed to extract the current profile. The current profile identifies the degree of coverage and the level of maturity of the controls. Using the result of the questionnaire it is possible, if deemed necessary, to slightly change the target profile to modify the priority levels

Evaluation

The assessment phase aims to analyze the gap between the current profile and the target profile, or multiple target profiles, defined in the contextualization phase. The gap analysis provides, through appropriate metrics, a measure of the organization's current cybersecurity posture relative to the defined target. It is important to note that the cybersecurity posture defined in the assessment phase is a relative and not an absolute measure. In fact, as the gap analysis is performed against a target profile, the results will differ depending on the profile being considered. The input of the assessment phase is the target profile, as defined in the contextualization phase, and the current profile obtained in the measurement phase.

The Results of the assessment process can be interpreted by the organization referring to different areas of interest that are identified in the methodology. An area of interest $A = (E, W)$ defines:

- A set of homogeneous elements E against which the assessment result is evaluated.
- An assignment of weights among the controls target profile $P\hat{}T$ and elements of the set E, using a matrix W, in which the element $W_i j$ is a value between 0 and 1 and represent how much the i-th control is relevant with respect to the j-th element of W.

Three areas of interest can be identified: (i) framework area, which allows evaluating how much the cybersecurity posture of the organization differs from the target profile; (ii) area of risk management, which allows evaluating how much the cybersecurity posture of the organization is in line with the risk mitigation objectives; (iii) compliance area, allow evaluating how much the current profile is compliant with current regulatory requirements in the context of the organization.

The metrics against which the assessment is evaluated are the following:

- Score: makes it possible to assess the degree of coverage in the implementation of measures needed to meet a given element of the set E.
- Maturity: allows us to assess the distribution of the degree of maturity with which the above measures are implemented across different controls.

The score can be computed by resorting to the following formula:

$$score(e_j) = \frac{\sum_{i=1}^{n} (x_i^A / x_i^T) \cdot W_{ij}}{\sum_{i=1}^{n} \cdot W_{ij}} \tag{3.2}$$

where e_j is the j-th element of E, $x_i^A s$ is the current coverage level of the i-th control (current profile), $x_i^T > 0$ is the desired coverage of the i-th control (target profile) and W_{ij} is the weight of the i-th control with respect to the j-th element.

The implementation maturity of the element e_j can be computed using the following formula:

$$m_j[k] = \frac{\sum_{i \in L_k(C)} \cdot W_{ij}}{\sum_{i=1}^{n} W_{ij}} \tag{3.3}$$

where $L_k(C)$ is the set of controls of the target profile to which in the current profile a level of maturity k were assigned.

By using these two formulas it is possible to analyze the result of the assessment and interpret the results to deploy an effective risk management strategy.

3.2 EU Risk Assessment Framework

In Europe, ENISA developed a risk assessment framework. The document "Methodology for Sectoral Cybersecurity Assessments" (SCSA Methodology) [52] was published in 2021 following, and building upon the European Cybersecurity Act (CSA) [53]. The proposed methodology has the objective of linking the process of cybersecurity and risk assessment with the requirements of the CSA in terms of cybersecurity certifications; providing a methodology that allows to interpret the result of the assessment with the certification process an organization may undergo following the ISO 27000 [46] and ISO 15408 [52].

The methodology first presents the concept of an ICT system as an ICT service that is supported by ICT products and ICT processes, and provides a distinction between two layers of an ICT system:

- ICT Infrastructure: which serves several markets, applications and end-users services. For example, an ICT infrastructure can be cloud services or mobile networks; infrastructure that provide some sort of service to the sectoral ICT system they support.
- Sectoral ICT Systems: which include all functions that are specific to the delivery of service to a particular market sector targeted at end-users. Usually Sectoral ICT systems rely on ICT infrastructure services for specific functions.

Moreover, the methodology identifies the composition, in terms of architecture and interaction among various components, of an ICT system as the basis for identifying possible threats and security requirements. This information should be identified as follows:

- Based on a risk assessment contextualized in the sectors of reference, stakeholders should identify security and assurance requirements for the reference ICT systems. They should also identify the infrastructure supporting the systems, its composition (ICT products) and the processes adopted.
- The requirements thus identified must be considered by stakeholders in defining the security problems of target ICT infrastructures, the security posture level (security functional requirements and security assurance requirements) necessary for target ICT infrastructures.

- The compliance with the requirements thus identified should be enforced to deployed ICT systems.

Analysis and identification of the context in which the assessment should or will be conducted are necessary for the efficient use of the SCSA Methodology. Identification of any information pertinent to the outline of the requirements and the risk assessment is crucial for the evaluation. What has to be provided includes:

- Description of the cybersecurity assessment scope and context.
- Documentation of involved stakeholders roles, including their responsibilities, and documentation of relevant business processes.
- Definition and documentation of business objectives and the requirements of identified processes.
- Detailed description of reference ICT system architecture and all operational processes to support the identified business processes.
- Identification and detailed documentation of *primary assets* supporting identified business processes; identification and documentation of ICT subsystems, processes, and services supporting primary assets (*supporting assets*).
- Identification and documentation of information needed to assess cybersecurity risk factors related and relevant to the context in which the assessment is made: threat and CTI information (i.e., attacker type, potential goals, motivation and potential)

The concept of primary assets and supporting assets is important in the methodology because it provides a way of linking together the cybersecurity risk assessment and the definition of the requirements for cybersecurity certification. The evaluation of cybersecurity risks depends on the sectoral stakeholders' objectives and the relevant business processes that might be impacted by security events. These factors should represent and take into account, in their definition, the application context, the business model, and the social and governmental context in which the organization moves and operates. The definition of certification, security, and assurance requirements, on the other hand, necessitates a technical and system design point of view. The identification of these requirements is done by focusing on the *supporting assets* of the analyzed system. These two different view points are bound together in the SCSA Methodology through the concepts of *primary information assets* and *primary functional assets*. *Primary assets* are information and functions that are relevant to the business goal; while *supporting assets* are

those that are supporting the primary assets and play a key role in the system. *Information assets* are basically data, where *functional assets* are software and hardware that allows activities to be carried out.

The SCSA Methodology defines the risk assessment as a function of the impact of an event and the likelihood of its occurrence. In order to provide more precise and detailed result the methodology contemplate and make use of CTI and attacker information.

Attacker information

The methodology provides metrics that can be used to identify possible attackers and their attack potential, in order to do so, it is suggested to leverage available CTI information to better capture attackers' information. Attackers' evaluation must take into consideration and based upon the identification of their motives.

- **Attacker motives**: The motivations that drive an attacker's actions; if the attacker is driven by a strong motivation then it must be considered that he/she will most likely be able to develop the necessary skills (technical and technological) to penetrate the system and perpetrate the attack. This evaluation must be conducted considering the particular context of the evaluated system or organization.
- **Attacker type**: if an attacker is part of a cybercriminal group, whether or not it is state-sponsored or has military capabilities, it has to be considered a highly skilled adversary that may have sufficient funding and skill to successfully perpetrate the attack.

The result of the attacker assessment should be a list of potential types of attackers, their possible motivations and targets and an estimation of their capabilities. Based on this information, following the methodology, attack scenarios can be identified and detailed depicting targeted assets of the ICT system with attackers' information and evaluation.

Impact estimation & Probability estimation

Based on the identified threat events, the stakeholders should be able to calculate the likelihood of each risk scenario occurrence as well as its impact on the business. The context in which the organization operate can strongly influence the impact and probability of event and should be carefully considered in the assessment of risk factors. In order to ensure comparability of results across sectors, in the methodology are defined some generic levels of impact that can be used in the risk assessment. These levels provide a qualitative measure of the damage a threat can cause to the organization and are represented as impact classes (IC). Moreover, the evaluation of threat events likelihood may come from different sources:

- The historical information related to past, similar, events in the contexts of the assessment.
- The analysis of potential attacker and their motivation and capabilities.
- The historical information of similar events in other context and sectors.

Also in this case, in the SCSA Methodology are provided qualitative levels for probability estimation to guide the assessment and ensure comparability of results.

Meta-risk classes

In the SCSA Methodology, the evaluation of risk levels is conducted resulting in meta-risk classes (MRC). These provide a qualitative measurement of the risk level that can be computed as a combination of the probability (likelihood) of a risk event happening and the evaluation of the consequences (impact) the risk event may cause. Moreover, some special cases are identified: (i) incidents with a very high impact on the system, which are considered catastrophic, should be assigned a high MRC regardless of the probability with which they may occur; (ii) adverse events that have consequences considered acceptable or of low impact should be assigned a high MRC in case these could lead to important consequences if they were to occur with a very high frequency.

Conduct Risk Assessment

The proposed methodology for risk assessment is composed of the following steps or workflows:

- Step 1: context establishment and business layer assessment.
- Step 2: primary asset layer assessment.
- Step 3: assessment of supporting assets.
- Step 4: supporting asset gap analysis.

The risk assessment is mainly conducted at the primary asset layer, while supporting assets inherit the evaluations and the risk identified from the linked primary assets. It is, anyhow, considered the possibility of re-assessing supporting assets in case a mismatch exists between the primary asset risk class and the actual risk class of the supporting asset.

In the first step, according to the SCSA Methodology, the goal is to gather and document information about the context of the assessment with respect to the perspective of what is relevant for consideration by the stakeholders (i.e., risk owners), as well as, contextual CTI

and attacker information. The activities to be carried out in the first step can be summarized as follows:

- Scoping the sectoral context: this task involves establishing the sectoral context as the foundation for conducting a sectoral assessment. The scope of the assessment should be determined by the sector's target audience, desired services, and use cases. In addition, the roles and responsibilities of sector stakeholders should be documented at this stage.
- Identification of required business processes: in order to support the targeted services and use cases, it is necessary to document the business processes. Furthermore, it is important to identify the sectoral stakeholders who contribute to or depend on these business processes.
- Identification of stakeholder objectives and requirements for each business process: in this stage, the organization should document the objectives and requirements of the stakeholders involved in each business process. The aim is to gather relevant objectives from various perspectives, including business, customer, societal, and governmental, depending on the stakeholders' roles. Additionally, the organization should address generic risk areas when defining the objectives and requirements for each business process.
- Classification of objectives and requirements: the objectives and requirements of stakeholders pertaining to the specific business processes may have implications for risk assessment, as well as for the characteristics of the cybersecurity scheme or for stakeholders' assurance needs.
- Identification and documentation of ICT subsystems or processes supporting business processes: the documentation of the business processes should serve as a basis for documenting the ICT subsystem or process that supports each process. This documentation should cover the entire architecture of the ICT subsystem, including all involved products, services, and processes. It should also include external components that contribute to the ICT subsystem. In the case of processes, such as the supply chain, the process flow as well as the supporting architecture and components should be documented.
- Identification of information and functional assets and related supporting assets: this step involves identifying and documenting the primary information and functional assets and their supporting assets for each ICT subsystem or process that has been identified. This includes identifying the most critical assets and an initial ranking of their relevance to stakeholders' objectives. In addition, potential attack scenarios and attackers capable

of conducting such attacks should be considered in the priority ranking. The sectoral stakeholders involved in the related business process should be consulted during this step to ensure alignment with their objectives.

- Documentation of context-specific CTI and relevant attacker types and their properties: The initial phase of contextual establishment concerning threats and adversaries should entail evaluating the risks, CTI data, and pertinent attacker categories for the scope and business processes described earlier. This evaluation should draw upon available sector-specific information, including threat landscapes.
- Documentation of relevant attack scenarios: relevant attack scenarios that could affect the identified primary assets should be identified and documented.

The second step in the methodology aims to identify relevant risk scenarios and assign *meta-risk classes* (MRC) for each risk scenario. The activities to be carried out in the second step can be summarized as follows:

- Identification of risk scenarios: risk scenarios will be defined based on the information gathered and documented in the previous step and by identifying attack scenarios targeting the identified primary asset.
- Assignment of probability level and impact class for each risk scenario: probability of occurrence and impact are assessed for the identified risk scenarios. The assignment of the probability level depends on the probability of occurrence of the attack corresponding to the identified risk scenario. The probability estimation must take into account several factors, from the attack scenario identified in the points above and from information provided by stakeholders about the organization's context.
- Assignment of the meta-risk class for each risk scenario: the identified risk scenarios, based on the assessment made, are assigned a corresponding MRC. At this point, the various risk scenarios must be documented in detail describing the identified and assigned rating classes (impact, likelihood, risk). The MRC assessment provides the yardstick for comparison among the various risk scenarios and can be used to identify a possible ranking; through this it is then possible for stakeholders to identify the threats that pose a higher risk and prioritize them. The meta-risk class assigned to a risk scenario is inherited by the primary asset and all its supporting assets.

The third step in the methodology aims to assess the supporting assets of primary assets identified in the previous steps. The main objective here is to identify requirements for

supporting assets in terms of certifications, security level and assurance. Activities to be conducted can be summarized as follows:

- Documentation of relevant parameters for each supporting asset and its intended use: the parameter relevant for the assessment of supporting assets are: the intended use, which is the role of the supporting asset in the infrastructure; the meta risk class for the intended use, this information is inherited from the risk scenario it is part of; operational environment and description of the architecture and implementation technology; specifications, standards and other requirements.
- Re-assessment of estimated parameters: given the connection between supporting assets and primary assets, usually the supporting asset inherits the risk class from the primary asset it is connected. But given that, sometimes, supporting assets may not be prone to attack as their primary assets, a re-evaluation of MRC for the supporting assets may be done to update this information.
- Definition of specific security and assurance levels: based on the result of the assessment carried out are identified security and assurance levels for supporting assets.

The fourth and last step in the methodology aims to analyze the result obtained in the step above in order to identify which are the necessary actions to be carried out in order to achieve the identified requirements for supporting assets and to evaluate the feasibility of those actions.

3.3 Threat Modeling Methodologies

There are several ways in which the process of threat modeling can be defined, depending on the domain of interest and the goal to be achieved [24]. A general definition, that captures a little bit what the process in general of threat modeling is, can be: “a systematic way to identify threats that might compromise security” [54]. Moreover, threat modeling has the following key advantages [55]: (i) it enables threat ranking by prioritizing the most critical ones and ensuring resources are dispersed efficiently to build and maintain appropriate defenses when used across the many phases of the product life cycle, from design to execution; (ii) by taking an iterative approach, periodically repeating the threat modeling process, it is possible to more or less ensure that the organization is always informed about what are currently the most important threats.

One of the fundamental steps, shared among most threat modeling methodologies, is the definition of the *Data Flow Diagram* (DFD) [56]. As the name suggests, it describes how data

are exchanged among the various components of the product under analysis, being it a software program or a whole IT infrastructure, indifferently. Analysts exploit data flow to understand which possible attacks threaten the system, how and which assets they impact, and so on.

Shevchenko et al. present in [57] an overview of the most widely used methodologies and approaches for threat modeling, whereas an overview of the available literature on threat modeling solutions can be found in the work of Xiong et al. [24]. Some of the most common ones include:

- **STRIDE**: is an acronym that stands for Spoofing, Tampering, Repudiation, Information disclosure, Denial of Service, and Elevation of privilege [58]. This methodology is widely used in Microsoft's Security Development Lifecycle (SDL).
- **Attack Trees**: is a graph based approach were for each identified attack is built a tree representing possible steps an attacker may undergo to successfully perpetrate the attack.
- **PASTA**: Process for Attack Simulation and Threat Analysis (PASTA) [59] is a risk-centric threat modeling methodology that takes a risk-based approach to identify and prioritize threats. It involves a seven-step process that includes scoping, data collection, threat identification, vulnerability mapping, risk rating, recommendations, and review.
- **Trike**: The Threat modeling Risk identification and Knowledge-based Engineering (Trike) [60] methodology is a knowledge-based approach that aims to systematically identify and evaluate threats based on attacker profiles, assets, and attack vectors.
- **TARA**: Threat Agent Risk Assessment (TARA) [61] is a threat modeling methodology that focuses on identifying and prioritizing threats based on the motivation, capability, and opportunity of attackers. It involves a four-step process: threat identification, threat analysis, risk assessment, and risk mitigation. These are just a few examples of the many threat modeling methodologies used in information security today, and the choice of methodology depends on various factors, including the scope and complexity of the system, the resources available, and the specific goals and requirements of the organization.
- **LINDDUN**: is a threat modeling methodology that focuses on privacy [62]. Threats are identified and categorized based on the type of privacy concern they pose to the system under analysis.

In the following, some of the most adopted threat modeling methodologies in industries and academic works are presented more in details.

3.3.1 STRIDE

STRIDE [58] is a threat modeling methodology developed by Microsoft and widely used in the software industry. The goal of this methodology is to identify potential threats to a software system and then apply appropriate countermeasures to mitigate these threats.

The STRIDE acronym stands for the following six categories of threats that are considered in the methodology:

- **Spoofing**: this is a threat where an attacker impersonates a legitimate user or entity in order to gain access to the system. This can be done through a variety of means, such as by stealing login credentials or by using a fake identity.
- **Tampering**: this is a threat where an attacker modifies data or code within the system in order to gain an advantage or to cause harm. This can include things like altering data values, changing system configurations, or injecting malicious code.
- **Repudiation**: this is a threat where an attacker denies responsibility for actions they have taken within the system. This can include things like deleting logs, altering timestamps, or using anonymous communication channels.
- **Information disclosure**: this is a threat where an attacker gains access to sensitive information within the system, such as passwords, credit card numbers, or personal information. This can occur through a variety of means, such as through network eavesdropping or through SQL injection attacks.
- **Denial of service**: this is a threat where an attacker attempts to disrupt or disable the system by overwhelming it with traffic or by exploiting vulnerabilities in the system's design. This can result in the system being unavailable to legitimate users or in reduced performance.
- **Elevation of privilege**: this is a threat where an attacker gains elevated access or privileges within the system that they should not have. This can occur through a variety of means, such as by exploiting vulnerabilities in the system's access control mechanisms or by taking advantage of programming errors.

The STRIDE methodology involves systematically analyzing a system or application to identify potential threats in each of these categories. The STRIDE methodology aims to provide guidelines for identifying threats by pushing analysts to question possible flaws in the system. Questions such as "How an attacker can change authentication data?", hence the simplest way

to apply STRIDE is to consider how each of the 6 threat categories in the model affects all the system components, their interconnection, and relationships. This analysis can be performed through a variety of means, such as through brainstorming sessions or through the use of specialized threat modeling tools (e.g., Microsoft Threat Modeling Tool [58]).

Once potential threats have been identified, appropriate countermeasures can be put in place to mitigate them. This can include things like improving access control mechanisms, implementing encryption, or adding monitoring and logging functionality.

An example of how STRIDE might be used in practice can be seen in the following scenario: suppose that a company has developed an e-commerce website that allows customers to purchase products online. The website includes a login system, a shopping cart, and a payment processing system. To perform a STRIDE analysis of this system, the following steps should be taken:

- Spoofing: potential threats in this category might include things like password guessing attacks, phishing attacks, or man-in-the-middle attacks.
- Tampering: potential threats in this category might include things like modifying product prices, altering shipping addresses, or injecting malicious code into the website.
- Repudiation: potential threats in this category might include things like deleting order records, altering payment histories, or using anonymous payment methods.
- Information disclosure: Potential threats in this category might include things like stealing credit card numbers, accessing customer contact information, or reading order histories.
- Denial of service: potential threats in this category might include things like overwhelming the website with traffic, performing distributed denial of service attacks, or exploiting vulnerabilities in the website's infrastructure.
- Elevation of privilege: potential threats in this category might include things like exploiting vulnerabilities in the website's access control mechanisms, gaining administrative access to the website, or exploiting programming errors in the website's code.

Once potential threats have been identified, appropriate countermeasures to be put in place could include, for instance, implementing two-factor authentication, using secure payment processing mechanisms, or implementing intrusion detection.

3.3.2 Attack Trees

Attack Trees [63] is one of the oldest and most widely used threat modeling methods. Initially applied alone, nowadays it is often found in conjunction with other methodologies such as PASTA or STRIDE. The proposed methodology requires the analysis of a system's vulnerabilities and potential attack scenarios by breaking them down into a hierarchical tree structure. For each of them, a graph is created that models how an attacker can achieve that goal, by describing the various steps of the attack and how it unravels within the system. This approach is used to identify potential attack paths and prioritize countermeasures based on their effectiveness in preventing attacks. The process involves constructing an attack tree that describes a system's security posture and identifying its weakest points or vulnerabilities.

The attack tree consists of nodes that represent various components of the system, such as the network, application, and database layers. Each node has a set of child nodes that represent possible attack scenarios, and each of these child nodes can have further child nodes, representing additional sub-scenarios.

The goal of describing attack trees is to identify the most likely and damaging attack paths through the system and to prioritize countermeasures that can effectively mitigate these attacks. This is achieved by evaluating the probability and impact of each attack scenario and assigning a score based on its severity.

Attack tree threat modeling is a powerful technique for identifying potential attack scenarios and prioritizing countermeasures that can effectively mitigate these threats. This approach has been widely used in various domains, including critical infrastructure, aviation, and cybersecurity, and has been shown to be effective in identifying the most critical attack paths and in prioritizing countermeasures that can effectively mitigate these threats.

Attack Trees are very simple and easy to adopt, but they require the involvement of analysts with high cybersecurity expertise and capable of clearly understanding the system composition and its related security concerns.

3.3.3 LINDDUN

LINDDUN has been proposed by Deng et al. and first published in 2011 [62]. It is a threat modeling methodology focusing especially on the privacy concept. The goal is to provide a methodology to analyze software systems to elicit the privacy requirements and select technologies to mitigate identified privacy-related threats and later identify the proper technologies to enhance privacy in the system.

In order to perform the threat modeling effectively, in the LINDDUN methodology some explanation and consideration about the concept of privacy within a system are done. Privacy may be divided into two types: *hard privacy* and *soft privacy*. Hard privacy is a data protection objective that refers to data reduction under the presumption that no private information is disclosed to outside parties. The goal here is to limit as much as possible the need to place some degree of trust in other organizations and limit as much as possible the data that are shared by the data subject. The threat model identified for *hard privacy* considers service providers, data holders and adversarial environments where threat actors are driven by the explicit intent to violate data privacy. In contrast, *soft privacy* is based on the assumption that the data subject no longer has control over his or her personal data, but must rely on data controllers who must then ensure competence and integrity in data management. Thus, soft privacy applies to the data management processes implemented by the data controller and defines policies, access control, and auditing as means of ensuring data security. The system model represented in the case of soft privacy is that of a subject who entrusts their data to a third party who is in charge of guaranteeing their security and protection. So the threat model used in this case is weaker and has different parties with unequal powers.

In the LINDUNN methodology, when dealing with IT systems, the classical security properties (also referred to as CIA triad) are considered for ensuring privacy: *confidentiality* (ensuring that data is disclosed only with authorized entities), *integrity* (ensuring protection of data from voluntary or involuntary modification), *availability* (ensuring that data are available, when requested, to authorized users), and *non repudiation* (ensuring that the entity that produced a certain piece of data cannot deny that it did so). Moreover, other privacy properties are also to be considered when dealing with threat modeling: *anonymity* (hiding connections between a piece of information and the identity of who created/shared it), *unlinkability* (hiding connection between two or more actions, identities and pieces of information), *undetectability* and *unobservability* (hiding user's activity), *plausible deniability* (other parties can prove someone has done something or produced information) and *forward security* (ensuring that encrypted data cannot be decrypted when at rest).

In the proposed framework all these privacy properties are considered and categorized as hard privacy properties, while policies, consent compliance and user content awareness are considered soft privacy properties.

LINDDUN is the combination of the privacy properties considered in the methodology and which threat have to be considered for threat modeling:

- **L**inkability of two pieces of information can allow an attacker to identify possible relationships between this information and use them to their advantage

- **I**dentifiability represents the ability for an attacker to identify the individual who produced a certain piece of information. In particular, Identifiability is a special case of Linkability where properties are involved that allow a subject to be identified in the system and associated with its data. Identifiability is a threat to both anonymity and pseudonymity.
- **N**on-repudiation, in some ways could be seen as a threat to privacy, and it allows an attacker to prove that a certain entity is aware of actions taken, data produced and so on. But it is important to ensure that an entity cannot deny having performed a certain action because system-generated evidence is present.
- **D**etectability means that an attacker is able to discern between information and normal noise. Thus, that it is able to identify information stored or transmitted within the system
- Information **D**isclosure considers those threats that go to reveal to a malicious actor the personal information of an individual in the system.
- Content **U**nawareness indicates the fact that a user is unaware of his or her information being shared within the system. This information can be exploited by an attacker to identify the user or access information that the user was unaware of.
- Policy and consent **N**oncompliance represents a discrepancy between what are the privacy policies that the system claims to implement and what are actually implemented. This may be due to the lack of some policies or the incorrect implementation; this leaves the system exposed to possible attacks and privacy problems even though nominally it should not be.

In the LINDDUN methodology the threat modeling process starts with a high level description of the system using a data flow diagram. Then, each of the DFD elements is associated and mapped to the provided privacy threats. The list of privacy threats is provided in the methodology.

The mapping of DFD elements to threats will generate a series of *privacy tree patterns*; these trees aim to identify possible privacy threats present in the system and describe how they may propagate into the analyzed system. Information is gathered about vulnerabilities that can be exploited to perpetrate an attack, the various predisposing conditions that can facilitate the implementation of a threat and so on. Finally, the system's privacy standards are drawn from instances of misuse. According to the privacy requirements, suitable privacy-enhancing methods are then chosen.

The LINDDUN framework does not cover risk assessment computation and has been developed independently of any risk assessment framework. This allows LINDDUN to be

adopted freely as a threat modeling methodology and use the result to evaluate the risk according to the framework/methodology of choice.

Chapter 4

State-of-the-Art

This chapter will present an overview of the academic literature related to the topics covered in this thesis. The various proposed approaches to vulnerability management, threat modeling, and risk assessment will be presented; with a special look at those proposals that use ontologies as a tool to support them.

There are several commercial tools that can support in the different stages of the cybersecurity risk assessment process. Most of provide access upon purchase of a license and, for obvious reasons, do not provide details on implementation choices or processes adopted, while for those tools that provide a free trial have major limitations in the functionality provided.

With regard to the vulnerability management process, the available commercial tools (e.g., Tenable[1], InsightVM[2], Intruder[3], and many others) are usually based on active scanning processes of the infrastructure to identify connected components and identify their possible vulnerabilities and guide mitigation activities through prioritization of the problems found. As far as threat modeling is concerned, there are few tools that allow automating the threat identification process, one of them is the Microsoft Threat Modelig Tool [58], which was briefly presented in Section 3.3.1. Regarding the risk assessment process there are both commercial tools (i.e., CRAMM[4]) and freely available tools (e.g., PILAR[5], EBIOS[6], ENISA tool[7]). These either rely on the use of questionnaires to perform the assessment (and automate it)

[1] https://www.tenable.com/products/tenable-io
[2] https://www.rapid7.com/products/insightvm/
[3] https://www.intruder.io
[4] http://www.insight.co.uk
[5] https://www.pilar-tools.com/en/
[6] https://www.ssi.gouv.fr/guide/ebios-risk-manager-the-method/
[7] https://www.enisa.europa.eu/cybersecurity-maturity-assessment-for-small-and-medium-enterprises/

or require extensive and highly specialized manual work by the user in entering and evaluating information.

Considering these factors, in this chapter it was decided to focus primarily on the academic literature both to highlight what are the concepts used in the literature to address the various stages of the risk assessment process and to evaluate the approaches that go toward the automation of cybersecurity risk assessment. In addition, since the approach proposed in this paper is based on the use of ontologies, it was decided to also focus on those solutions that make use of ontologies in this context.

4.1 Cybersecurity Ontologies

As will be presented later in this chapter, several ontologies have been proposed in the literature to support security management, both to provide a shared and standardized tool for representing data, but also as a reasoning ground for inferring and deriving useful information. The work of Adach et al. is presented an overview of the related literature [64].

In the sequel, we present two of the main, and probably most notable, examples of cybersecurity ontologies. These two ontologies have been the seed for the work presented in this Thesis and some of the concepts used have been extracted from these two.

4.1.1 Unified Cybersecurity Ontology

The *Unified Cybersecurity Ontology* [65] is an extension of the Intrusion Detection System ontology that wants to provide a common structure to describe the cybersecurity domain. The main idea of this project is to try to change the approach of the cybersecurity standards from a syntactic representation to a more semantic representation. To do so, a large number of existing standards and ontologies have been studied and reviewed and the most common ones have been selected to be integrated with UCO. In particular, UCO is intended to be the semantic version of *STIX* [66] and it includes references to many external standards, such as *CVE* [31], *CWE* [32], *CVSS* [67], *CAPEC* [34], *CYBOX* [8], *KillChain* [9] and *STUCCO* [10]. Another important external reference integrated into UCO is the *Linked Open Data* [68]: a dataset of general world knowledge in the Linked Data format used in this context to support different use cases.

[8] http://cyboxproject.github.io
[9] https://stixproject.github.io/documentation/idioms/kill-chain/
[10] https://github.com/stucco

The UCO is made of classes, terms and rules. The top-level classes are:

- **Attack**: it describes cyber attacks;
- **Attacker**: it describes the adversary;
- **AttackPattern**: it describes from the attacker point of view the most common ways of implementing an attack;
- **Consequences**: it describes the possible effects of an attack;
- **Exploit**: the description of an individual exploit;
- **Means**: it describes the ways in which an attack can be performed.

Then there are other classes resulting from *STIX*[11]. STIX is a structured language for describing cyber threat information that consists of 9 key entities and the relationships between them [66]:

- **Observables**: describe stateful properties or measurable events in information technology;
- **Indicators**: combine observable patterns with contextual information to identify a possible behavior;
- **Incidents**: describe instances of specific adversary actions;
- **Adversary Tactics, Techniques, and Procedures**: describe attack patterns, malware, exploits, kill chains, tools, infrastructure, victim targeting, and other methods used by the adversary; in UCO are represented with the *TTP* class;
- **Exploit Targets**: describe vulnerabilities, weaknesses, or configurations that might be exploited;
- **Courses of Action**: describe actions to address a threat that may be preventative or reactive;
- **Campaigns**: describe sets of incidents and/or TTPs pursuing a shared intent;
- **Threat Actors**: describe identification and/or characterization of the adversary;

[11] https://stixproject.github.io/about/

- **Reports**: collect related STIX content and give them shared context; this class is not present in UCO.

So as far as attacks and threats are concerned, the information is written and shared using a structured language and this is a great advantage for interoperability.

The same applies to vulnerabilities represented with classes *CVE, CWE, CVSScoreType* and *OSVDB*. The implementation choice made is to represent external references as parent classes; this allows ontologies to expand independently and makes them easier to manage.

UCO is at the moment the most comprehensive and exhaustive ontology for cybersecurity related information and it has been mapped to all the most common publicly available ontologies, but for the purpose of this Thesis one fundamental aspect is missing: the description of the infrastructure. In fact, the focus of UCO is to give an overview about all general concepts in cybersecurity, including possible attacks, attack patterns, means and consequences, but only very few classes are dedicated to the description of the infrastructure. The classes *Address, File, Hardware, NetworkState, Product, Software* and *System* are present but they do not allow for the detailing of the different components of the infrastructure and the relationships between them.

The use cases for which UCO is designed are for example the discovery of all the vulnerabilities of a certain vendor or the search for the least vulnerable software of a specific category, so the idea of collecting knowledge from different sources remains very important, but it is not intended for finding out all the vulnerabilities of specific infrastructure.

4.1.2 IoTSec Ontology

Another important existing ontology in the domain of cybersecurity is the *Internet Of Things Security Ontology* [69]. The goal of the authors of IoTSec is to have a resource that helps gain a big picture of information security in the Internet of Things field.

IoTSec introduces the concepts of *Asset* and *SecurityMechanism* that were not present in UCO and that help outline an overview of a single infrastructure. In fact, IoTSec was created to ensure that companies in the IoT field use their devices in the most secure possible way and to help them identify vulnerabilities and criticalities. Being IoT-oriented, many subclasses of *Asset* are very specific, such as *UMTSTechnology, LTETechnology* and *BluetoothTechnology*.

The class *SecurityMechanism* represents as well a positive aspect of this ontology because it allows representing concepts such as cryptographic algorithms, firewalls, protocols and other mechanisms that are being used. By making use of the reasoner, it is possible to infer for

example the necessity to introduce a new security mechanism to protect a specific asset or to find out that a security mechanism already present is not sufficient to defend a *SecurityProperty*.

The negative aspect of IoTSec is that the knowledge of the classes *Vulnerability* and *Threat* is static: it does not come from external sources and it must be managed and maintained directly when implementing the ontology. In particular:

- **Vulnerability**: it contains a set of sub-classes listing the possible vulnerabilities for the different technologies (for example *BluetoothVulnerability, WebVulnerability, SoftwareVulnerability*) and for some of these sub-classes a set of individuals is specified (for the *WebVulnerability* some of the instances are *BrokenAuthenticationSession, SensitiveDataExposure*);
- **Threat**: it contains the threats corresponding to the vulnerabilities previously defined (for example *BluetoothAttack, WebAttack*); also in this case some instances are defined such as for the case of *WebAttack* where there are *BruteForce, BufferOverflow, XSS*.

So, the idea is to already have a complete and detailed list of the possible vulnerabilities and attacks for the cybersecurity field in the ontology. This is helpful when performing assessment of an infrastructure, but it makes it more difficult to always have an up-to date list; as a matter of fact, the updates must be performed manually by the user because the language used is not standard, so it can become a long process and can produce errors.

4.2 Vulnerability Management

Considering the continuous growth in the complexity of both information systems and security information, it becomes more and more necessary to provide solutions that facilitate the user in the management and use of these large and complex knowledge-bases.

New vulnerabilities are discovered almost every day and collected in large vulnerability databases (CVE, NVD, and so on) that are very complex and difficult for security analysts to manage by hand. It is then important to provide solutions and tools to support, and automate to the maximum extent the process of vulnerability management.

Vulnerability management refers to the process of identifying and classifying vulnerabilities in a system. These are analyzed and assigned a rating that allows their prioritization; in addition, possible countermeasures for remediation and mitigation are identified. The process of vulnerability management is to be considered, and is certainly more effective in this way, as

a cyclical process that aims to keep the system's vulnerabilities under control and identify new ones [70].

4.2.1 Ontology-based Solutions

The authors of [71] presented an ontology-based semantic wiki, VuWiki, to provide a structured representation of vulnerabilities. The paper is not focused on cybersecurity, but rather it presents a generic approach to vulnerabilities and assessment in a broader sense. The aim of VuWiki was to provide the means to review and represent the results of vulnerability assessments and was developed as a reference system for describing the process. The ontology was implemented in a semantic wiki that allows classification and annotation of vulnerability assessment; it was not intended and designed to provide a general, holistic model of vulnerabilities, but to provide a reference system for security analysts and to provide a structured and easily accessible knowledge-base for information extraction. The ontology and semantic wiki were created with the intention of creating a framework focused on solutions and, via the establishment of semantic rules, facilitating comparisons between various vulnerability assessments and applicable concepts and methodologies. In practice, the design of the ontology aims to answer, and thus consequently represent, 4 basic questions: (i) vulnerability to what?, (ii) vulnerability to what?, (ii) what framework was used to perform the assessment?, (iv) what methodological approach was used to perform the assessment?

The work of Mozzaquatro et al. [72] presents and proposes a framework that uses the IoTSec ontology to support during the design phase of a system to identify the technology to be implemented and, given a list of requirements (i.e., some type of requirement for data encryption or communications) generates pieces of code that can be used by developers to integrate them into the developed software and protocols. In addition, the proposed framework also aims to support security analysis of a system in the operational phase. The various monitoring tools contribute to the population of the ontology with alerts generated by the analysis of activities occurring in the system. At this point the identified alerts are mapped to vulnerabilities and threats present in IoTSec to identify remediation actions to be implemented. It is therefore necessary that there is a mapping between the alert generated and the content of the ontology otherwise nothing is provided.

A similar, more recent work, is the one proposed by [73]. The paper presents the developed ontology, Cybersecurity Vulnerability Ontology (CVO), which provides the information representation model needed for vulnerability management. After that, a system, Cyber Intelligence Alert (CIA), is presented, which by exploiting the CVO ontology provides cybersecurity alerts

about vulnerabilities and countermeasures. The authors' goal is to design an ontology to organize vulnerabilities, including those from unstructured sources such as social media or blogs, so that analysts can better and more easily analyze vulnerabilities and their severity. By combining the vulnerability concepts outlined by the NIST, CERT/CC, and CVSS, the Cybersecurity Vulnerability Ontology (CVO) offers a formal knowledge representation of the vulnerability management domain. The CVO also maps the notions of vulnerability that were taken from the well-known microblogging website, Twitter. Next, the CVO is used to create the Cyber Intelligence Alert (CIA) system, an ontology-based alert system that issues cyber alerts based on underlying defined rules. The CVO may be applied practically as the vulnerability management domain's common vocabulary, while the CIA system can be used to generate warnings about emerging vulnerabilities, exploits, patches and so on. The CIA system, at the highest level, gathers vulnerability data from a variety of sources, including CVE, NVD, Twitter, and vendor websites. The CVO, which represents the vulnerability domain, is used by the CIA system to integrate and reconcile disparate data. After that, the CIA system improves the CVO by adding new concepts needed to send cyber warnings and creates the Cyber Intelligence Ontology. (CIO). The CIO acts as a knowledge-base for alert generation.

The work of Wang et al. [74] proposes an ontology-based approach that enables the analysis and assessment of the security posture of software. The ontology, called Ontology for Vulnerability Management (OVM) [75], developed for vulnerability management and analysis, allows for measuring the trust level of a given software whose analysis is to be done based on the specified security requirements and from the characteristics of the software itself. The proposed approach is a holistic approach that wants to consider the software in its entirety for security assessment instead of performing the analysis only on individual components of the software. The ontology, to make support and perform the security assessment, integrates information from standards such as CVE, CPE, CWE, CVSS, and CAPEC and is supplemented with automated tools developed ad-hoc to support the proposed methodology. The developed ontology is populated with vulnerability information taken from the NVD database and relates it to the categorization of software products; a categorization that is done, and described in the ontology, based on the functionality a software may have. IT administrators can gauge the severity of the target software product in their IT environment with the use of vulnerability ontologies and security metrics. Additionally, users may query the OVM to determine connected goods and gather relevant vulnerability data for each product. According to the environmental security indicators, we will determine the environmental security ratings for each product. The ontology is designed to accommodate information gathered from vulnerability databases (i.e., CVE, NVD, and so on). The ontology, once populated can be used by the user to search for vulnerabilities of a given product; the vulnerabilities found for the given product are then used

to compute the trustworthiness of the product itself. In order to compute the security posture of a given software, the authors suggest using the CVSS (Common Vulnerability Scoring System)[67] of each of the target product vulnerabilities and combining these values together to compute an overall score of the security of the target product. The ontology must be populated by hand.

In [76] authors proposes a system based on the developed ontology, Dynamic Cybersecurity Ontology (DCO), which aims to support dynamic analysis of IT systems security. The ontology developed allows the mapping of results from scanning services such as Shodan [77] or Censys [78], these provide results that, thanks to the ontology, are represented semantically and related to each other. The relationships among the extracted vulnerability information are organized by referring to common IT systems security concepts that are also represented in other ontologies to allow for better interoperability and easier integration. The classes included in the ontology are: (i) organization, which represents the main element, and represents the target organization on which the scanning is being done and for which the results are being collected; (ii) service, which represents the particular service being scanned; (iii) vulnerability, which represents the identified vulnerability and refers to the CVE database and the Exploit class; (iv) exploit, which represents the given identified exploit of the vulnerability and allows specifying the exploit code, type, and result; (v) CVE, which represents the CVE entry related to the vulnerability and is characterized by the CVE-ID, description, references, and generally all the information that can be extracted from the CVE knowledge-base; (vi) product, which represents the scanned product with version and vendor information.

A similar approach is proposed by Arnaert et al. [79]. They propose an ontology for modeling vulnerabilities for IoT objects that are based on search engines such as Shodan and Censys to gather the required information to populate the ontology. The proposed approach aims to reduce the number of aggregated information and increase the relevance of information resulting from the scanning in order to provide users a more easy access to discovered vulnerabilities and information. Thus, the proposed work aims to support the use of analysis tools such as Shodan and Censys to simplify the interpretation of the results. The proposed ontology allows the representation of different types of vulnerabilities divided into categories (e.g., input validation, privilege escalation and so on). The ontology has, therefore, the goal of allowing a user to use information received from the search engine tools, to search for known vulnerabilities saved within the ontology. The work is interesting, but the authors do not provide information on the composition of the ontology. The issue of population of the vulnerability ontology is not mentioned, which is left to the user to take results of the scanning and populate the ontology.

The work of Kim et al. [80] proposed an ontology-based knowledge base that leverage on represented APT attack instances and system domain knowledge to recommend security requirements. Three parts make up the proposed knowledge-base: (i) ontology for APT knowledge: APT knowledge is the knowledge gleaned from APT case study analysis. The attack element in APT knowledge is defined as the attack tools applied in the situation and the attacker's behaviors. It is also possible to describe the social engineering techniques the attacker employs to pose as the target in order to spread malware. (ii) ontology for general security knowledge: general security knowledge includes aspects needed to establish security needs as well as security requirements themselves. This knowledge covers aspects of security such as security objectives, requirements, and countermeasures, as well as aspects of systems such as assets and people that need to be assessed for risk. It also includes elements that define threats such as attack vectors and vulnerabilities. (iii) domain-specific ontology: contains system-related knowledge specified by a company or organization. This comprises a list of resources such as assets, people, system goals, and business goals that are specific to a certain domain.

The proposed solution has no automation of any kind; the ontology is expected to be populated with the necessary information by the user. Both in terms of attacks, vulnerabilities, and assets. The part of the ontology concerning APTs tries to provide the necessary entities for their modeling (the type of attack and the goal, understood as business, economic, or political), while for vulnerabilities and assets two entities related to each other are used. No distinction is made between asset types etc. The knowledge base can then be used to, following a risk assessment phase, provide security requirements to be implemented related to the identified attacks and vulnerabilities. Neither the risk assessment methodology nor the reference requirements are given in the article.

The work of Steele [81] proposes an ontology that does not represent the concept of vulnerability per se, but rather aims to support reasoning over the system under analysis and is composed of two main parts: (i) the Access Point class which represent entry point in the system where users can interact with, these entry points are identified as those point in the system that upon receiving users inputs they change the state of the system;(ii) the Asset class which represent the component of the system that is valuable and needs to be protected, assets are categorized in software, hardware, and information. The objective of the designed ontology is to, therefore, guide the user in understanding the possible impact a vulnerability may have on the system under analysis. The ontology has to be populated by the user representing the various assets, their access point and how these may give access to other assets in the system. Once the ontology has been populated, there will be different chains representing the various

path that lead to access certain assets. Using these paths it is possible to understand what an attacker may access given a certain vulnerability, and so on.

An ontology-driven framework for analyzing the impact of cyberattacks designed specifically for the analysis of cyber-physical systems is presented in Venkata et al. work [82]. The framework allows for the identification of vulnerabilities within the system under analysis and provides a semantically reasoned mechanism for identifying the impact they may have on the physical system. In addition, downstream of the identification of vulnerabilities, the system also provides an indication of possible countermeasures that can be taken both in the physical world and in the cyber mode and possible changes to the system design that can increase the security of the system. The proposed ontology provides the knowledge-base necessary (Ontology Knowledge Base, OKB) for the operation of a framework developed by the authors themselves. The ontology serves as a means of data collection and organization, while other tools in the framework (VULCAN [83] and NEMESIS [84]) are used to extract and analyze information. OKB plays a central role within the VULCAN framework by capturing various critical public data feeds of IT products vulnerabilities, attack, and mitigation information. The vulnerabilities information extracted refers to publicly known vulnerabilities that affect the system under analysis. Then, the information gathered about the products is analyzed by the NEMESIS tool, which allows a vulnerability to be associated with the reference STRIDE category. In this way it is then possible, by referring to STRIDE, to propose remediation actions to be taken; for example in case of elevation of privileges it is recommended to upgrade or patch the system, the use of authentication against spoofing and so on.

4.3 Threat Modeling Automation

In this section is proposed a brief presentation of the major solutions aiming at automating the threat modeling process, or part of it.

4.3.1 General Solutions

Frydman et al. [23] developed a tool, named *AutoSEC*, which aims to automate the threat modeling process. The authors focus on the key aspect of introducing automation in order to reduce the need for highly specialized personnel to perform this type of operation. AutoSEC includes several parts. Regarding the threat modeling part, the DFD generated using the Microsoft TMT tool is used as input. After that, two data structures are used: *identification trees* and *mitigation trees*. These data structures, built from databases such as CAPEC and CWE

contain the information to identify a threat within the DFD and the possible countermeasures that can be used related to the identified threat. The tool then automatically derives the threat model using the data structures to map the threats on the system using the DFD.

Two works extend STRIDE targeting specific domains.

In the former, Casola et al. [85] propose a system for automatic threat modeling for edge computing scenarios. The work is designed and tested as an extension of the Microsoft Threat Modeling Tool. A library of elements is provided for the description of the IoT/Edge computing world, where the various components are characterized according to different aspects, such as location within the infrastructure, type of data processed, and so on; information that will be used in the threat modeling phase. The system modeling proposed is based on three main asset categories: *physical/virtual processing nodes*, *software components/modules*, and *communication channels*. Data, instead, are divided into *user-related data*, *environmental data*, and *service data*. As far as threats are concerned, the authors constructed a threat library using mainly scientific articles as sources. In the threat modeling phase, particular attention has been paid to the aspect of false negatives: in fact, according to the type of element of the system some threats, not possible in that specific case, are not considered, to highlight only the significant results.

In the latter, Flå et al. [86] propose an extension to the Microsoft Threat Modeling Tool (TMT) for Smart-Grid threat modeling. A template is created to be imported into TMT, which consists of nodes (in this case referred to as stencils) to describe a Smart Grid context and a list of threats to be considered. The categorization of the threats is based on STRIDE, and for each category, some possible threats are identified in a Smart Grid context. The threats considered are very general, a choice also made to avoid having too many false negatives.

4.3.2 Ontology-based Solutions

In this section we propose a brief presentation of the major solutions that explicitly exploit ontologies.

One of the most recent and most noticeable solutions concerning ontology-driven threat modeling is the work performed by the OWASP foundation[12]. Among their numerous projects, the *Ontology Driven Threat Modeling* Framework automates the threat modeling process using ontology to collect all the information concerning the security of a specific domain[13]. The project is still in progress, but the idea is to build a base threat model ontology and,

[12] https://owasp.org
[13] https://owasp.org/www-project-ontology-driven-threat-modeling-framework/

from it, to provide more domain-specific ontologies and ontology-driven threat modeling rule engine. The *Base Threat Model* is an OWL ontology [87] that contains classes and properties representing the components of the infrastructure, the threats, the mitigations, and the properties that bind all of these classes. Another important piece of information that this model allows representing is the *DFD Diagrams* [88]. Starting from the *Base Threat Model*, it is possible to build more *Domain Specific Models* that contain the typical components and the threats and countermeasures of a specific area [89, 90]. The project focuses primarily on cloud-based infrastructures.

Välja et al. [30] proposed an ontology framework to improve automatic *Threat Modeling* that aims to address two problems: the lack of domain knowledge and the mismatched granularity. The goal of the proposed ontology is to provide a means for the unambiguous representation of information from a variety of sources; in fact, the information (i.e., names) used to describe the components of a system may vary depending on the interlocutor. They consider four different types of data elements: *software products*, *operating systems*, *other types of applications* and *data flows*. For each of these, key concepts (i.e., software name, software version, and so on.) are identified by the authors and represented in the ontology to ensure that the information collected is complete and with the appropriate level of detail.

Sabbagh et al. [91] provide a socio-technical framework to support the analysis of possible threats that may be generated in the supply chain of software products for a company. The framework relies on an ontology whose function is to provide a comprehensive and formal knowledge-base and modeling language for the system under analysis and the related threats. The proposed solution does not involve any kind of automation, but is intended to provide a tool that analysts can use to reason, holistically, about the supply chain security issues and how to address them effectively. Threat modeling focuses on issues that refer to ineffective coordination between security measures used by the supplier and the customer.

Kamongi et al. [84] propose a solution, named *Nemesis*, for automating the threat modeling to risk assessment process that uses an ontology as the knowledge-base for describing and modeling threats in the system. The proposed solution consists of different automation software that computes the results, while the starting knowledge-base is that provided by VULCAN, a risk assessment framework, previously developed by the same authors [83], which requires a good deal of manual interaction in the assessment process. The authors then use the ontology underlying Vulcan and automate the interaction with the framework; the results extracted from the knowledge-base, in the form of vulnerabilities, attacks, and countermeasures are then used within Nemesis to generate various analytics.

Salini et al. [92] propose a solution aiming to provide a means of predicting web application attacks given the domain-specific threats and vulnerabilities present. An ontology that models the possible domain-specific threats and vulnerabilities of web applications is used as the basis of the solution. A third component of the ontology models the possible attacks by combining the threat and vulnerability information present. The derivation of possible attacks is done through the use of inference rules and automatic reasoners. The proposed solution is not designed for threat modeling per se, but rather to allow an analyst, given a vulnerability (e.g., cross site scripting), to understand what the possible attacks are (e.g., XSS Attacks). At this point, the system also proposes some possible countermeasures to be taken.

The solution presented by Ekelhart et al. [29] aims to simulate the possible impact that a given threat event may have on the analyzed infrastructure. The proposed ontology therefore is designed to allow the description of threats of various types, including both cybersecurity threats and physical threats (i.e., fires and natural disasters). The threat ontology follows a taxonomy defined by Landweher [93]. It is not particularly detailed, but very general and describes only the concepts of the CIA triad, without going into detail about specific threats. Assets description is not very detailed and allows a very generic description of the infrastructure in question. The solution then allows to simulate threats impact on the company assets considering scenarios with and without countermeasures in place. The example discussed does not involve cyber threats, but rather a case of a fire within the company infrastructure.

Luh et al. [94] propose an automated system to understand the impact an attack can have on a particular infrastructure, in terms of assets involved. The results can be used to plan appropriate countermeasures to be taken. The solution consists of an ontology used to describe APT-type attacks, broken down into their phases. The authors use as a basis for modeling the attacks that of Cyber Kill Chain by Hutchins et al. [95] and MITRE ATT&CK [96]. Interestingly, an automatic reasoner is used as a method for extracting information items from the ontology, once the infrastructure assets are defined it is then possible to infer information about possible attacks through the reasoner.

In the work of Manzoor et al. [27] a tool for automating the threat modeling process for cloud infrastructures is proposed. The work is based on an ontology used to describe the different components needed for security analysis and threat identification. One part of the ontology is in charge of describing the security requirements of the infrastructure, one part for describing the vulnerabilities, and one part maps the vulnerabilities with the consequences they may have on the requirements, organized according to the CIA triad. The main drawback of the solution is that the threats are identified as the known vulnerabilities entered, so a vulnerability will correspond to a threat in the final matrix, thus going to limit the effectiveness of a threat

modeling operation, which should provide an overview of possible threats in part by abstracting from the known vulnerabilities so as not to risk leaving parts of the system uncovered during the analysis.

4.4 Risk Assessment

Over the years and especially recently, several solutions to risk assessment have been proposed that seek to support analysts in this type of activity. Some of the main solutions and approaches proposed are collected in this section.

4.4.1 General Solutions

Based on the technological and application domain considered several risk assessment solution have been proposed in literature, ranging from application in cyber physical systems to IoT and more general IT systems.

In the work of Tantawy et al. [97] authors proposed a model-based risk assessment for cyber physical systems. They proposed an approach to analyze and design cybersecurity solutions for a given CPS.

The first phase of the risk assessment concerns the identification of physical threats which are then used, leveraging hybrid automation, to identify potential hazardous states of the system adopting a mathematical model of the physical system analyzed. The authors describe possible attacks as connected graphs, starting from the identified entry point (i.e., compromised corporate PC) to the target of the attack (i.e., compromising valve functionalities) all possible paths and actions to achieve the goal are described. The process adopted in this case is similar to attack trees and is performed resorting to network and data flow models to describe the threat scenario leading to identified hazards. The attack graphs are then used as guidelines to conduct penetration testing activities to assess vulnerabilities of the system. The result of the penetration testing is used to compute the probability of an attack happening, and this information is combined with other, such as attackers' capabilities and impact to compute the overall cyber risk. The computed risk level is then used to plan remediation.

Another example is the work of Bobolt et al. [98] in which they proposed a method for cybersecurity risk assessment of ship systems. The proposed approach builds on top of and improves the Cyber-Preliminary Hazard Analysis (CPHA) method [99], and is developed to support cyber risk assessment design improvement of the analyzed system. The developed

method, CYRA-MS, consists of different phases. Starting with the system modeling are then identified possible attack scenarios which are clustered in specific attack groups and are analyzed one at a time. During the identification of possible attacks also system vulnerabilities are identified resorting to existing vulnerability databases. Then, based on the attack group and vulnerabilities are identified potential attack types and potential consequences on the system components. Follows the estimation of consequences severity and attack likelihood based on the different parameters characterizing an attack. Finally, the risk is computed and improvements for the systems are identified.

The work of Farzan et al. focuses on cybersecurity risk assessment in power grids [100]. The paper proposes a methodology to identify critical assets both at power grid and substation levels and to assess related cybersecurity risks. The methodology is based on adopting two models. The first model is used to identify and assess substations criticality and rank them accordingly. The first model starts from the business mission and goes down to the network nodes to identify vulnerabilities and calculate the risk. Identification of vulnerabilities is performed through a questionnaire submitted to users. The second model instead goes into the details of the substations networks identifying components and related vulnerabilities to compute the risk level. Also in this case the vulnerability assessment is performed using a questionnaire. The risk methodology proposed combines assets reliability, vulnerabilities and attack cost to evaluate risk.

Moving to the IoT domains, the work of Nurse et al. [101] presents an overview of some of the issues that standard approaches to risk assessment may have when applied to an IoT ecosystem. The authors present the contrast between static standard methodologies, which are primarily designed for systems that are stable (in their composition) over time, and the IoT world, which usually sees very sparse changes in infrastructure composition. Given that risk assessment should be performed periodically, the fact that in the IoT world changes are very rapid leads the authors to indicate the need to propose continuous and automated risk assessment approaches and the need for modeling and simulation tools that can support analysis activities.

Pandey et al. presented an automated threat-based risk assessment methodology for smart home scenarios. Their work is part of the GHOST project [102], the risk assessment is based on predefined risk levels and the authors adapted the *Negative to Positive Model*[103] for the identification of the risk levels. The identification of risk exposure at different risk levels is performed by resorting to different influence factors. The influence factor considered are: (i) physical, the tangible assets that compromise the network; (ii) customer/user; (iii) societal, the perception that the society has regarding a component of the network; (iv) reputational,

perception that competitors, suppliers and stakeholders have regarding a component of the network; (v) intangible/logical, intangible assets, such as data generated in the network; (vi) legal and regulatory, potential sanctions resulting from a breach. Then are evaluated the costs of mitigating the risk or not both taking into account the cost of adopting security measures and the cost (in terms of impact) of deciding not to address the risk. The information on identified risk level, cost/benefit of taking or not a mitigation action and the different influence factors are used to compute the risk level and monetary impact to plan appropriate risk management decisions.

Moving into the general IT domain, Gonzalez-Granadillo et al. proposed a risk management framework tailored for privacy that takes into consideration the GDPR regulation [104]. The paper proposes AMBIENT, a toolkit for cybersecurity and privacy-related risk management. The proposed solution leverages several sources of information, from questionnaires to collected data from monitoring sensors (SIEM), to assess risk in an organization. Risk analysis is carried out based on the information collected through the sensors installed in the system for capturing events and the documentation provided by the organization regarding the data processing processes adopted. Moreover, risk assessment also take into consideration ICT infrastructure assets and their relationships with data sources and data subjects. The risk assessment procedure provides also a list of suggested mitigation and security mechanisms that the security administrators could implement.

The work of Biswas et al. [105] proposed a risk assessment framework that leverages the content of hacker forums to evaluate attackers' potentials. They proposed a text-mining based approach organized in two main steps. First is the risk assessment process that uses machine learning to identify hacker expertise analyzing messages exchanged in hacker forums. In this step, they analyze different features such as keywords, sharing of attachments and sentiments to distinguish between expert hackers and novices while linking this information to attack types. The second step consists in evaluating threats consequences, where financial impact is computed for each attack scenario based on the relationship between the identified attack and the hackers capabilities regarding it. The financial impact is the raked on a scale to prioritize mitigation strategies. According to the claims of this study, it is possible to determine the level of hacker expertise on a dark forum. Such results have intriguing implications for organizations trying to figure out who these harmful individuals are and these kinds of data can be used by security analysts to follow evolving or often mentioned attack paths, keep track of the activities of specific hackers, and more.

Kure et al. [106] proposed a framework called i-CSRM to support the cybersecurity risk management process; the proposed framework through an assessment of the effectiveness of

implemented security controls and through machine learning techniques aims to predict different types of threats to which the system is potentially exposed. The i-CSRM framework leverage a number of different open source security information (CWE, CAPEC), while for security control they refer to the CIS critical security control. Additionally, for risk assessment and management they leverage CTI information and machine learning models. The results obtained in the assessment phase combined with a database containing data on activities and procedures for effective risk management are used to provide the organization with the information needed to support the development and improvement of the risk management process and prevent possible attacks. The framework is built to support analysts in the various step of risk assessment; asset criticality is supported by fuzzy set theory to provide a mathematical estimation, for vulnerabilities criticality evaluation a reference table is proposed to be used as guidelines in the process and threats are identified resorting to the CAPEC classification and association with IoC information contained in the ATT&CK knowledge base is performed. These operations are manually performed by security analysts and the framework provides guidelines, reference tables and so on to guide through the process. Results are then used to compute risk assessment, in the first phase using machine learning and past incident data, risk types are identified; then each risk type is evaluated by resorting to information about threats and vulnerabilities. To each risk type are associated controls to adopt for mitigation.

4.4.2 Ontology-based Solutions

The research field of risk assessment the use of ontologies, as a means of providing a formal and semantically rich data representation as well as for inferring information, has seen several types of applications.

The work of Sales et al. [107] proposes an ontology to address the problem of the conceptually unclear notion of risk. They proposed COVER, an ontology that tries to relate, and integrate with each other, the different concepts of risk that may be found in the published research literature and the concept of value. The proposed ontology links the concept of risk, understood as the actions that can lead to the realization of a threat, with the concept of value so that an economic value can be assigned to a potential risk. The first similarity that can be identified between the concept of risk and the concept of value is that nothing is innately valuable or dangerous; rather, both concepts are goal-dependent. Things don't merely have value; they have value for someone. Similarly to this, things are only in danger from the view point of someone. The ontology formally characterizes and integrates three different concepts of risk: (i) as something measurable, thus risk as a quantitative concept; (ii) as a set of events

leading to consequences, thus the concept of risk as something impacting the given goals; (iii) as something assignable, thus risk as a relationship.

Building on COVER, Oliveira et al. [108], developed an ontology (ROSE, Reference Ontology for Security Engineering) for risk treatment and management. The goal of ROSE is to model security mechanisms as ways of dealing with and mitigating risk. Authors introduced several concepts to deal with security mechanisms and risk treatment; they introduced the concepts necessary to describe the goals (what a security mechanism wants to protect) of a security mechanism, the link between a risk event and the capabilities of a security mechanism to identify which risk event can be mitigated. Moreover, they introduced the concept of protection trigger, which describes what is the event that triggers the activation of the security mechanism and so on. The ontology thus allows reasoning about which security mechanisms can be adopted to mitigate the risk events identified.

Moving specifically to cybersecurity ontologies, several solutions have been proposed to support in different part of the risk assessment process.

The work of Grigoriadis et al. [109] proposes a solution to support risk-related information collection operations in cyber-physical systems. The proposed ontology models and unifies with a formal representation information collected from different sources considering security knowledge-bases (structured data), information from security reports (unstructured or semi-structured data), and information collected from network monitoring systems that can be used in the system to be analyzed. In addition, this information is also enriched by considering normal users of the system and possible threat actors. The ontology provides concepts to model a cyber-physical system (devices, networks and human actors) and concepts to model vulnerabilities, threats, attacks and threat actors. To model this part of the ontology they based the entities around the metrics required for evaluating the CVSS score of a vulnerability and threats are identified as the vulnerabilities in the system. The proposed solution is designed to support risk assessment by providing a way of collecting information from several available sources and correlating this information together.

Two recent works tried to address the problem of contextual information for risk assessment, providing an ontology for collecting alert information from SIEMs to provide a real time risk evaluation and management. The work of Riesco et al. [16] was the first proposal, later extended by Sánchez-Zas [110] to cover the IoT domain.

The proposed solution includes the translation of the STIX v2.0 standard into an OWL ontology and the rules necessary to calculate risk levels from incidents detected and represented in the STIX standard. The identification of relationships between an asset and collected IOC, threats or security events is performed resorting to an implemented graph data model. The

developed framework then allows, CTI information to be represented in TTP patterns, which are then used to share rules for detection and calculate the associated risk on the infrastructure. In practice, the proposed solution takes as input the possible incidents that are identified in the infrastructure and imports them into the developed ontology to perform the necessary analysis through the defined rules. This allows having a dynamic and near real time risk analysis based on the identified incidents in order to propose appropriate risk management actions.

A similar approach has been proposed by Arogundade et al. [111]; also in this case, risk analysis is done from incidents detected by an IDS to support risk management. The ontology-driven knowledge-base generated by the probing agent is used to classify the security risks detected by the IDS, and long-term frequency probability is used to determine the likelihood of real-time threats happening. Basically, the system receives input information of detected attacks from an IDS and analyzes it to assess the risk on the system under analysis and identify possible security measures to be implemented. Attacks identified by the IDS are categorized by the system according to STRIDE and information to be monitored for further investigation is identified. Of the identified threat, the level of risk it poses to the system and possible actions to be taken to mitigate this risk are also calculated. The risk is calculated based on the information provided by the IDS and information from similar threats previously discovered and stored in the knowledge-base. While the activities to be taken to mitigate a threat are either provided based on past actions taken for similar threats or a qualitative assessment is provided with high-level actions to be taken.

Another similar work is the one proposed by Khaleghi et al. [112]. The authors propose a cyclic model that describes the interactions that occur on the network under analysis between network components and attackers. Basically, based on the actions that occur on the network, the model enters or remains in a cyclic state that depends on the calculated level of risk. A first cycle, *Deterrence Cycle*, represents the state of the network where the risk is tolerable, security solutions are implemented, anomalous activities identified and/or the presence of vulnerabilities are under control, and the risk does not exceed the imposed threshold (i.e., an attack costs more than the benefits it brings to the attacker). A second cycle, *Resilience Cycle*, represents the state of the network where an attack is in place, but the network is able to resist and mitigate the consequences of the attack. If the attack is blocked or mitigated, the state of the model reverts to the previous state of deterrence. The last cycle, Survival Cycle, represents the case where an attack has been successful and defenders try to contain it by keeping the network functioning. If the activities to contain the attack are successful, then the system returns to the resilient state, otherwise if the defense fails and one or more components of the network are unusable then they must be replaced. To calculate and determine the state of the network, information is analyzed that is collected from the various installed sensors that monitor the

activities occurring on the network. To do this the ontology is provided a mapping between what are identified as instant events (a measurable event that can cause a major change in the network, i.e., the connection of a new device) and the related security attributes (activities that represent, i.e., threat, incident, countermeasure and so on).

4.4.3 Considerations

In summary, several works and solutions have been proposed in the literature over the years that aim to support security analysts in their risk assessment journey: from vulnerability management to threat modeling to end with the whole risk assessment process.

Regarding the vulnerability management and identification part, several solutions have been proposed where ontologies play the role of means by which to organize, represent, and link information together. This allows for easier management of vulnerabilities through the use of automation and the possibility of inferring additional information from imported ones. The proposed solutions, at the automation level, however, mainly refer to the use of automatic network scanning systems, which, in turn, are based on asset discovery systems, without going to provide a detailed representation of the analyzed system. The use of network scanning is yes a great help, but there is a risk of incomplete results as these tools cannot identify everything.

The solution proposed in this thesis bases the extraction of known vulnerabilities from a detailed description of the network composition, supporting the user in modeling the system through an ontology. In this way, even devices not directly accessible through the network can be considered in the identification of possible known vulnerabilities.

As for the threat modeling part, the works proposed in the literature mainly focus on specific vertical domains, making them hardly reusable in different contexts. In addition, they lack standard representations of models and data used in the various stages of the process. In addition, the inference rules supporting the reasoning task during the threat identification process are generally targeted to specific vulnerabilities and attacks. The threat modeling process is often done starting from the vulnerabilities identified in the system; though this is certainly an effective method and one that highlights known problems within the system, there is a risk of overlooking problems that may be present but not known and discoverable only through further analysis. Furthermore, the proposed approaches only automate part of the threat modeling process, still requiring human interaction in the definition of threats and resorting to external tools, in a not integrated way. Finally, aspects related to supply-chain threats are rarely considered.

The approach proposed in this thesis provides a standard metamodel to describe a complete ICT infrastructure, establishing a formal and common vocabulary for all the involved parties; threats are already categorized and formalized in the designed ontology, starting from common threat databases. In addition, a well-defined set of inference rules are provided to describe the threat modeling logic and to map infrastructure components to threats using ontology automatic reasoners. In addition, supply chain related security concerns are taken into consideration. The proposed approach can be easily extended, and no external tools or coding are required.

Finally, regarding risk assessment, solutions proposed tend to provide automation only for the result computation part, while still depending heavily on human interaction for threat modeling and other elements of the risk assessment process. Moreover, recent solutions tend to have a data-driven approach to risk assessment exploiting information from system monitoring solutions. While data-driven approaches are certainly effective and can gather more easily contextual information, they start from the assumption that a monitoring system is deployed within the ICT infrastructure.

The approach proposed in this Thesis builds on top of information automatically extracted in the vulnerability management and threat modeling phases and leverage this information to assess the risk of identified threats.

Chapter 5

Proposed Solution

This chapter presents the proposed solution and contribution of this Thesis work. The work presented mainly refers to the publications [113, 2] regarding the part related to vulnerability management and threat modeling. In this chapter, the developed ontology will first be presented in the detail of its various parts, after which the proposed approaches for collecting security information needed to populate the ontology will be presented and how this information is used in the various parts that make up the risk assessment process (identification of vulnerabilities and threats and assessment of the risk associated with them).

5.1 Architecture

As anticipated in Chapter 2, the goal of this work and the proposed solution is to provide a tool that can be used to manage the cybersecurity of an ICT infrastructure and that can support the organization or entity that uses it in performing some of the activities necessary to assess the security posture. In addition, the data collected and information obtained can be used within the risk management process to be able to make informed decisions.

A high-level view of the proposed solution is presented in Figure 5.1. Conceptually, this consists of the following elements: (i) a metamodel described using an ontology, (ii) a modeling tool, (iii) a reasoning engine, and (iv) the resulting knowledge-base.

The **ontology metamodel** describes the concepts and provides the vocabulary needed to describe the case study. Using an ontology for this purpose brings several benefits: it both allows to communicate and share a formal and structured representation of knowledge, and it

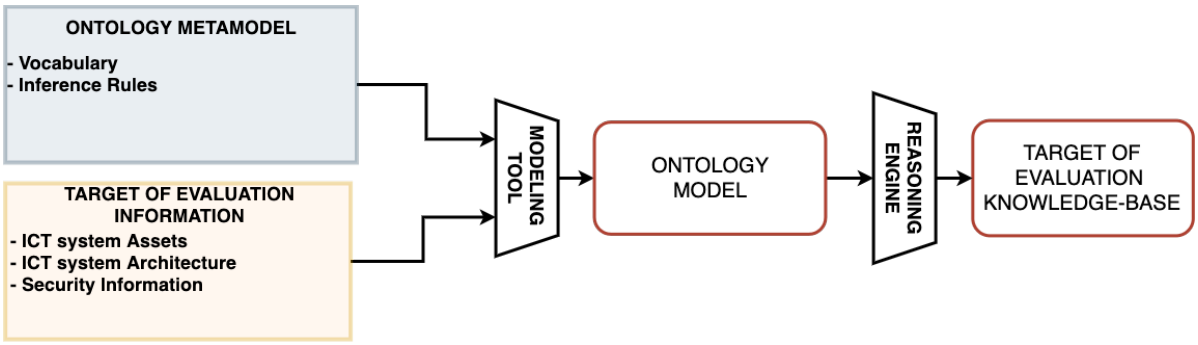


Fig. 5.1 High Level View of the Proposed Solution

provides a means to collect and represent information in a machine-readable and semantically rich format so that it can be conveniently analyzed by automated tools.

OWL2 [114] was chosen as the reference language for the design and development of the ontology. This is the latest and current version of the Ontology Web Language (OWL): a modeling language is used to express metadata such as concepts, relationships between these concepts and their features in a way that is both machine-readable (for interoperability) and human-readable. OWL is based on the Resource Description Framework (RDF) semantics [115] but is an extension of it, to increase expressiveness. The World Wide Web Consortium (W3C) proposes RDF and OWL as Semantic Web standards for ontologies. The first major difference between these two standards is that OWL adds more vocabulary to describe properties and classes: for example, it introduces the possibility of expressing cardinality (i.e., "exactly one"), or more specific relationships between classes (i.e., disjunction) and many others. The other difference is that RDF performs no consistency check on the logic of statements, so anything can be said, whereas OWL imposes strict constraints on logical consistency. For example, in RDF an object can be both an instance and a class, whereas in OWL this is not allowed. The third difference is that OWL has a rich variety of annotations that makes it very easy to link different ontologies together. OWL thus allows for greater expressiveness and tighter control of the concepts expressed.

The **modeling tool** allows the user to enter the necessary information, or “populate”, the ontology defined by the metamodel. This operation makes it possible to collect the information needed to create the model of the system under analysis. Protégé [116], a tool developed by Stanford University that allows building ontology-based applications, was used as the modeling tool in this work. It embeds the necessary features for describing ontology, populating it, and applying automatic reasoners.

The **reasoning engine** represents the component that, starting from the provided model and the defined inference rules, infers new information, which may be new relations or computation of values. In this work, SWRL [117] was used as the inference rule definition language, while Pellet [118], which supports both OWL and SWRL syntax, was used as the automatic reasoner. The definition of SWRL rules and the application of the reasoner were always done using the Protégé tool. When inference rules become ineffective due to their inherent expressiveness limitations, one must rely on additional external tools that allow automation of the analysis.

SWRL allows writing rules expressed through the OWL concept to reason about the content of the ontology (i.e., the "individuals"). These rules allow new concepts, or knowledge, to be generated from what is already contained within the ontology.

The **resulting knowledge-base** stores all the information already in the model, the information describing the target of evaluation in all its aspects, and the new information inferred through the reasoning engine.

Going into more details about the proposed architecture, Figure 5.2 presents the various components that are actually part of the proposed solution, their role and purpose.

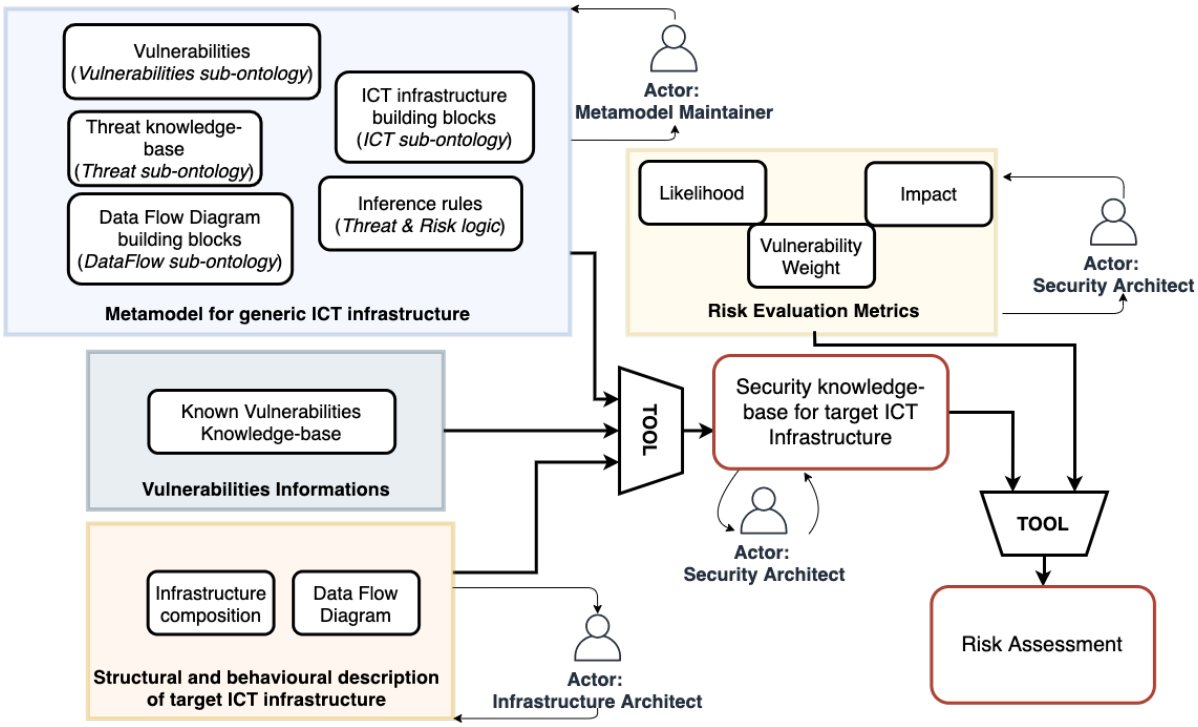


Fig. 5.2 General Architecture of the Proposed Approach

The first component is the `metamodel for generic ICT infrastructure` which is conceptually divided six five parts, identified as sub-ontologies, that are part of the same

ontology metamodel: (i) Vulnerabilities, (ii) Threat knowledge-base, (iii) ICT infrastructure building blocks, (iv) Data Flow Diagram building blocks, (v) Inference rules, and (vi) Risk metrics. These are necessary to model all the information needed to collect and evaluate the various security information related to the infrastructure under analysis.

- `ICT sub-ontology`: this part of the ontology is in charge of modeling the reference ICT infrastructure. It provides the vocabulary for describing a generic ICT infrastructure to be used when creating the model.
- `Vulnerabilities sub-ontology`: this part of the ontology is in charge of modeling the vulnerabilities present in the ICT infrastructure under analysis. It provides the vocabulary needed to describe the vulnerabilities, their characteristics, and the concepts necessary for the representation of how the vulnerabilities affect the infrastructure under analysis.
- `Threat sub-ontology`: this part of the ontology is in charge of modeling cybersecurity threats that may target the referenced ICT infrastructure. It provides the vocabulary necessary for the representation of threats. In this case, as it will be analyzed in more details in Section 5.4, the model is generated considering threats in general, since the identification of which threats may afflict the infrastructure under analysis is done through the use of inference rules.
- `Data Flow sub-ontology`: this part of the ontology is in charge of modeling the Data Flow Diagram (DFD). It provides the vocabulary needed to describe the interaction between and with the infrastructure components.
- `Risk sub-ontology`: this part of the ontology is in charge of modeling the information needed for cybersecurity risk assessment. It provides the vocabulary necessary for the representation of the information needed for cybersecurity risk assessment for the infrastructure under analysis.
- `Threat logic`: this part of the ontology collects the inference rules used to derive new information with respect to what is stated in the created model (i.e., threat identification).

The metamodel is maintained by a *Metamodel Maintainer*, i.e., a person or a team responsible for creating and maintaining the provided metamodel.

The second component, `vulnerabilities information` of the proposed solution, includes the information about the known vulnerabilities , extracted form various knowledge bases available on the Internet, such as the CVE, CWE, and NVD databases. These information

items, as will be shown in Section 5.3, are collected from these databases to identify the set of already known vulnerabilities present in the infrastructure components under analysis and mapped to them.

The third component of the proposed solution consist in the `structural and behavioral description of target ICT infrastructure`, which contains information about the infrastructure under analysis. This is provided by the *Infrastructure Architect*: the person responsible for the infrastructure under analysis. She/he is in charge of designing, by exploiting the rules described in the metamodel, the target infrastructure, by creating a model containing assets, connections, and interactions among the various components. The provided description is then used to populate the ontology metamodel to describe the target ICT infrastructure to be analyzed.

These three components are provided as input to a modeling tool that allows the defined metamodel to be populated with the information necessary to create the reference ontology model. The inference rules defined in the metamodel are applied to this model in order to extract additional information regarding the security of the system (i.e., threat modeling), thus going on to generate the *cybersecurity knowledge-base*. This represents a picture of the situation of the analyzed infrastructure containing the list of assets, structural and behavioral descriptions of the infrastructure, known vulnerabilities of infrastructure components, and the identified possible threats. The cybersecurity knowledge-base then provides the basis for further analysis of the security posture of the infrastructure that can be done by the *Security Architect*.

Thus, the fifth component of the proposed solution is the information needed for risk assessment, the `risk evaluation metrics`. These, as will be shown in Section 5.5, can be provided by the *Security Architect*, but can be collected as well from online databases, such as CAPEC. They are used, with the help of a tool, to provide an assessment of the risk associated with the threats identified in the system (contained in the cybersecurity knowledge-base). The result of this operation is the *Risk Assessment* that can be used by the *Security Architect* in support of cybersecurity risk management to identify or prioritize the actions necessary to mitigate the identified risk or to lower it to levels considered acceptable.

A sequence diagram of the actions to performed to exploit the proposed solution is in Figure 5.3.

The first step consists of the creation of the metamodel, the creation of the vocabulary needed to model a generic ICT infrastructure and security information, the creation of the threat knowledge-base needed for the threat modeling operation, and the definition of the inference rules used for the automation of the security assessment process.

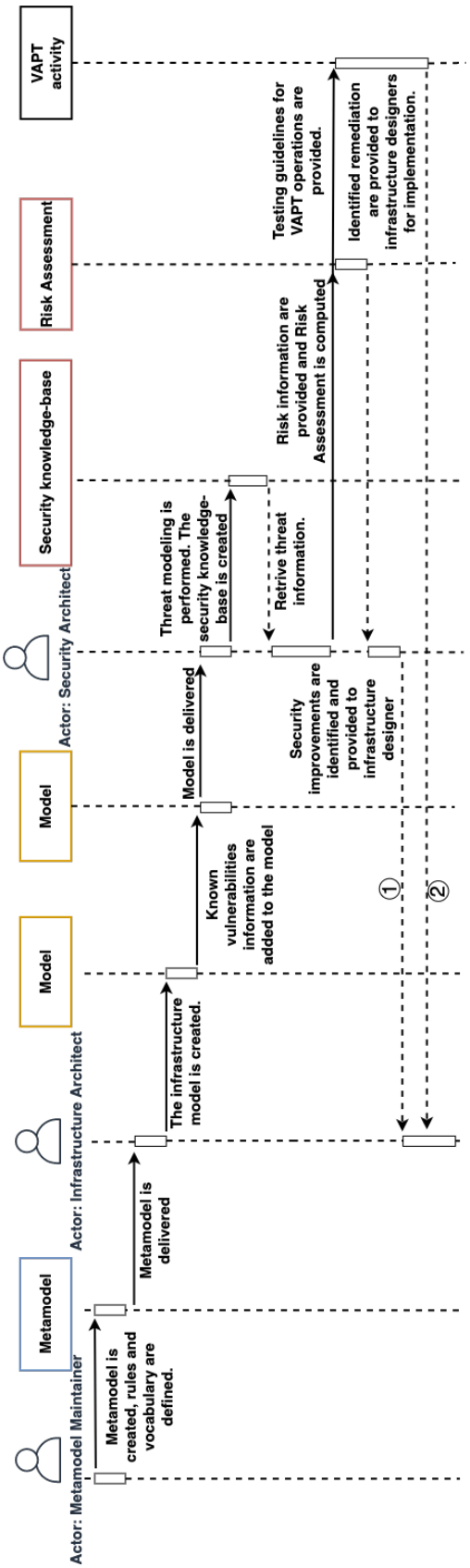


Fig. 5.3 Sequence Diagram of Proposed Approach Activities

The metamodel is then shared with the *Infrastructure Architect* who is in charge of populating the metamodel ontology with the information regarding the infrastructure under analysis. After generating the infrastructure model, information about known vulnerabilities related to the components of the modeled infrastructure is collected through the use of external tools. This information is entered into the ontology model.

The model thus generated is then shared with the *Security Architect*, who has the ability to revise and supplement the vulnerability information with the inclusion of vulnerabilities not known a priori that, for instance, could have been identified through VAPT operations. Contextually, threat modeling is performed with the information in the model. The possible threats to the infrastructure are then identified and a threats-to-assets mapping is done in an automated way, using the inference rules defined earlier. The result of these operations (identification of known vulnerabilities and threat modeling) generates the *security knowledge-base* that collects security information about the analyzed ICT infrastructure.

Once the *security knowledge-base* has been generated, the *Security Architect* can review it and enrich it with the information needed to assess the cybersecurity risk. The proposed approach is that of a threat-driven risk assessment; thus, each identified threat is associated with the information needed to assess its probability of occurrence and possible impact on the analyzed infrastructure. Again, this information is gathered by referring to available external knowledge-bases (e.g., CAPEC), but the *Security Architect* still has the ability to review and modify it, as needed. After that, it is possible to move on to perform the risk assessment. The purpose of this operation is to assign a rating to the identified threats and consequently sort them according to their dangerousness.

Moreover, two use case examples are shown in the figure. ① represents as an example the fact that the output of the risk assessment process can be used for Risk Management operations, such as the identification and prioritization of mitigation to be fielded and implemented in the ICT infrastructure.

The second case presented ② refers to the use of cybersecurity knowledge-base information, specifically known identified vulnerabilities and threat modeling that can be used to perform VAPT operations. In fact, on the one hand, it is possible to provide the list of known vulnerabilities, associated with all the related information such as examples of exploits or contextualization of the vulnerability within attacks. On the other hand, it is possible to use threat modeling as a guideline to understand which aspects of security should be further analyzed by activities aimed at identifying additional vulnerabilities not yet present in the vulnerability databases.

5.2 Ontology

The word *Ontology* comes from the Greek word *òntos*, which is the present participle of the verb *to be*, and the word *logos*, which means *logical discourse*; hence ontology literally means *discourse on being*; it is, in fact, a branch of philosophy that studies being and its categories.

This term has become commonly used in the field of computer and information science when talking about knowledge representation. In this sense, an ontology is a formal description of a set of concepts belonging to the same domain. These concepts are divided into *classes* and, for each class, the hierarchical relations linking it to the others (superclass-subclass) are established. The features and attributes of the different classes are called *properties*, the objects of the class are called *individuals* or *instances* and, for each class, the range of values that its *instances* can take is defined [119].

All the assumptions within an ontology are represented explicitly and this allows the ontology to be interrogated with automatic reasoning tools in order to obtain further information to enrich the knowledge base; it also allows checking the consistency of data and their properties.

Another important aspect of ontologies is that they create a common language and a knowledge that can be shared, integrated, and reused. In the cybersecurity field, in particular, many different sources produce data and information using different standards and languages. Having a deeper integration of data could allow changing the approach that most systems have towards an attack from reactive to proactive or even predictive [65].

To provide a common lexicon background, useful for the fruition of the chapter, some terms used in the description and design of ontologies are here explained [120]:

- **Entity**: the term entity is usually adopted to denote class, property, rule, and individual constructs in the ontology.
- **Class**: classes represent the concepts the ontology wants to model in the domain of interest; they are sets that contain individuals. Classes can express a hierarchy, or taxonomy, using subclasses. Subclasses specialize their superclass, and an individual of the subclass is also an individual of the superclass.
- **Property**: properties represent relationships. There are two types of properties, “object properties” and “data properties”. Object properties represent the relationship between two individuals. Data properties represent the relationship between an individual and a data type (i.e., a string, an integer, and so on) that can be used to assign values to individuals.

- **Individual**: individuals represent objects in the domain of interest, the *things* in the domain of interest used to populate the ontology. Individuals are associated with the class of reference. For example, in the context of this work, the class `Device` is used to represent IT devices and so, for instance, an individual of that class could be a router of the infrastructure to be modeled.

- **Rule**: rules are statements in the form of an if-then sentence that can be used to infer information or knowledge from the knowledge contained in the ontology.

With regard to the design of the proposed ontology, well-known and widely accepted design principles were followed [121, 122]. These principles are as follows:

- **Clarity**: an ontology should be able to clearly convey the definitions' intended meanings. Definitions ought to be impartial. The reason for describing a term may come from societal circumstances or numerical needs, but the definition itself should be unrelated to either of these contexts. "Formalism" is a tool for achieving this. A description should be expressed using logical principles whenever possible. A full definition is favored over a partial definition whenever feasible. Natural language should be used to describe every term.

- **Consistency/Coherence**: therms used in the ontology must be unambiguous and logically consistent. An ontology must support conclusions that are compatible with the definitions in order to be coherent and consistent. The defining principles should at the very least be rationally coherent. The ontology is illogical if a conclusion drawn from the principles conflicts with a description or informal illustration provided.

- **Conciseness**: according to [123] ontology must not have redundant or unnecessary concepts or attributes.

- **Extendibility**: a common lexicon should be used as a starting point for an ontology. The ontology must be designed by providing a representation of concepts that, where necessary, can be extended and specialized by the addition of further details. In other words, it should be possible to define new concepts for particular purposes based on the vocabulary already in use without having to change the meanings of the terms and concepts already in use.

- **Correctness**: according to [124] an ontology must model, through the defined classes and relationships, correctly the real-world concepts that are to be represented.

- **Minimal ontological commitment**: A minimum ontological commitment should be necessary for an ontology to support the planned knowledge-sharing activities. An ontology, in representing the concepts to be modeled, should make as few assumptions as possible with respect to the context being modeled, so as to allow the user the flexibility to specialize and implement the ontology as deemed necessary. Ontological commitment can be minimized by trying to define only high-level concepts and those concepts that are fundamental to knowledge sharing; thus, without overspecializing the ontology and limiting its expressiveness but trying to provide the user with the ability to generate different models as needed.

Trying to follow the above listed principles, some considerations were made, resulting in specific design decisions:

- **Clarity**: with respect to clarity, the concepts described in the ontology and the terminology used, refer to fairly standardized definitions shared by the community. Regarding the concepts of vulnerability, threat and risk and their properties, reference was made to official sources such as resources from MITRE, NIST and so on. Regarding ICT infrastructure, for the high-level concepts, reference was made to the TOGAF ArchiMate standard [125, 126]; regarding the specialization of these concepts into sub-concepts, widely shared and unambiguous terms were used or, again, reference was made to well known sources such as UCO [65].

- **Consistency/Coherence**: also in this case, the concepts described in the ontology refer to official sources so as to ensure disambiguation. In addition, during the validation phase of the ontology, the concepts needed for modeling an ICT infrastructure were refined to make them more precise and consistent.

- **Conciseness**: in this case, during the development and validation of the ontology, an effort was made to minimize redundant concepts. In fact, as will be shown later, there are no redundant concepts in the ontology (e.g., there are no classes that have subclasses in common, i.e., the classes graph is actually a tree).

- **Extendibility**: regarding extendibility, since its creation, the proposed ontology has been created with this concept in mind. In fact, the ontology presents a hierarchy in the definition of concepts ranging from high-level definitions toward the low-level ones, thus allowing the ontology to be easily extended by integrating and extending existing concepts with new information or technological domains. In fact, the goal of the ontology is precisely to make it possible to extend it to make it more and more complete and precise.

- **Minimal ontological commitment**: Regarding this last point, however, in the specific case of the proposed ontology, it was a trade-off work with respect to providing a generic ontology and narrowing the available modeling options. In fact, given the goal of the proposed solution and again referring to the ICT infrastructure part of the ontology, the design choices and specialization of the ontology are dictated by the need to have the information categorized in a certain way in order to then be able to apply automatic reasoners and infer the necessary information. As an example, one could leave the generic concept "*asset*" to describe an infrastructure component to allow for the most flexibility, but this would not have allowed for proper *asset-threat* mapping, so it was necessary to specialize the "*asset*" class into subclasses that better identify its type and characteristics.

As anticipated in Section 5.1, regarding the inference rules used to extract new knowledge from the ontological model, SWRL [117] was used.

The SWRL language allows users to write rules expressed in terms of OWL concepts for reasons about OWL individuals; these rules can and are usually used to infer new information from what is already present in the ontology knowledge-base.

In SWRL [117] each rule is composed of an *antecedent* (body) and a *consequent* (head), each of them consisting in a set of *atoms* (which can be also empty); informally, a rule may be read as meaning that if the antecedent holds (i.e., the stated condition is true), then the consequent must also hold. An empty antecedent is treated as always holding true, while an empty consequent is treated as holding false. Atoms can be of form $C(x)$, $P(x,y)$, $sameAs(x,y)$, $differentFrom(x,y)$, or $builtIn(r,x,...)$ (built-in refers to atoms provided by SWRL, i.e., mathematical operations) where C is an OWL description or data range, P is an OWL property, r is a built-in relation, x and y are either variables, OWL individuals or OWL data values, as appropriate. Hence, an atom $C(x)$ holds if x is an instance of the class description or data range C, an atom $P(x,y)$ holds if x is related to y by property P, an atom $sameAs(x,y)$ holds if x is interpreted as the same object as y, an atom $differentFrom(x,y)$ holds if x and y are interpreted as different objects, and $builtIn(r,x,...)$ holds if the built-in relation r holds on the interpretations of the arguments. Atoms may refer to individuals, data literals, individual variables or data variables. Variables are treated as universally quantified, with their scope limited to a given rule. Only variables that occur in the antecedent of a rule may occur in the consequent.

SWRL also provides a "human readable" syntax to represent and write rules, which greatly reduces the complexity of working with SWRL. In this syntax, rules can be expressed as:

$$antecedent \Rightarrow consequent$$

According to this syntax, as an example, a rule expressing the relationship with parents and their brother to imply, or derive, the uncle, can be expressed in the following form:

$$hasParent(?x, ?y) \wedge hasBrother(?y, ?z) \Rightarrow hasUncle(?x, ?z)$$

which means that if $?x$ (Alice) has as a parent $?y$ (Bob) and $?y$ (Bob) has a brother $?z$ (Charlie) then $?z$ (Charlie) is the uncle of $?x$ (Alice).

Figure 5.4 depicts the complete designed ontology; each piece will be analyzed and presented in more detail in the remainder of this chapter.

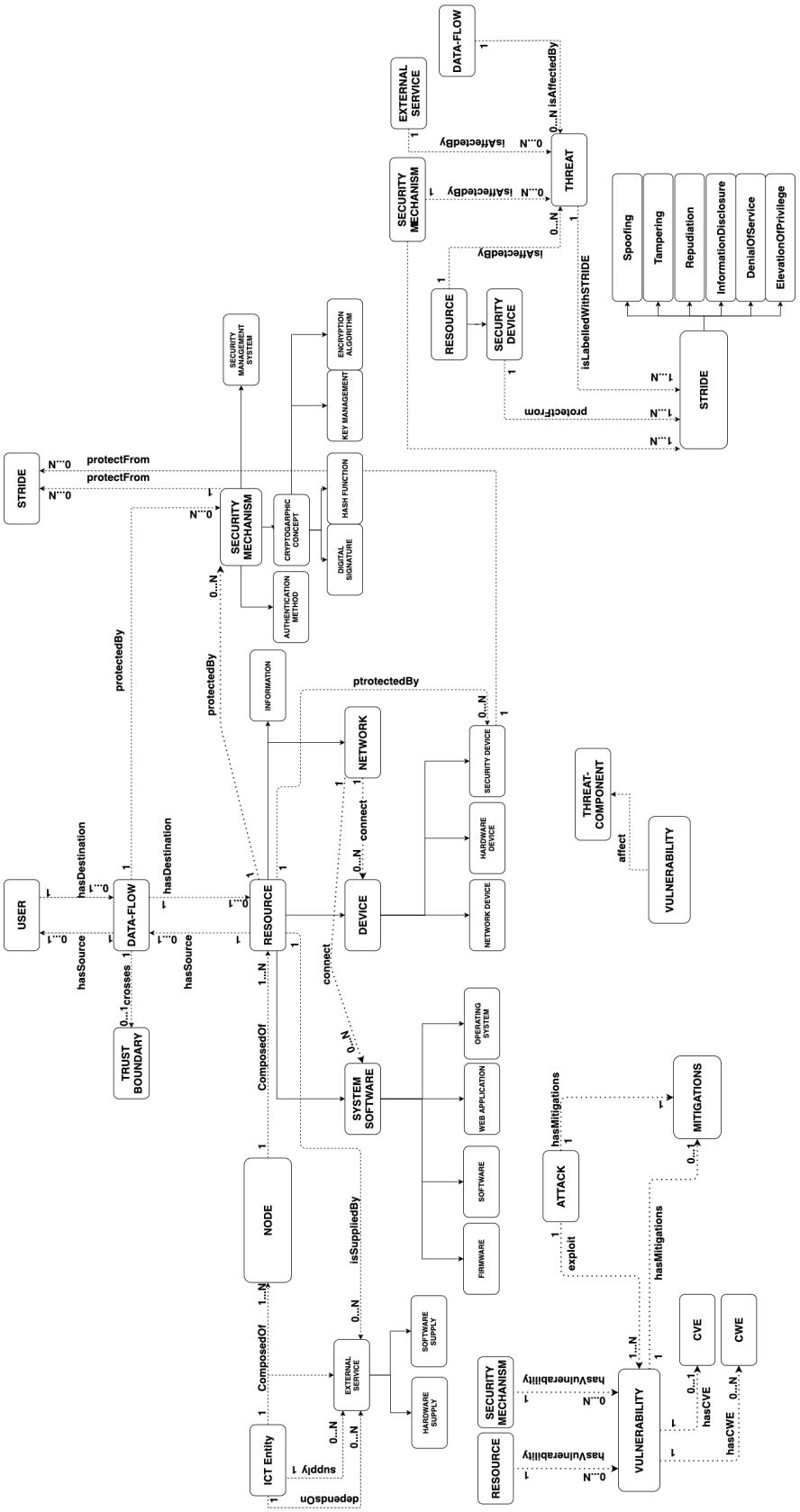


Fig. 5.4 Complete View of the Proposed Ontology

5.2.1 Modeling ICT Infrastructures

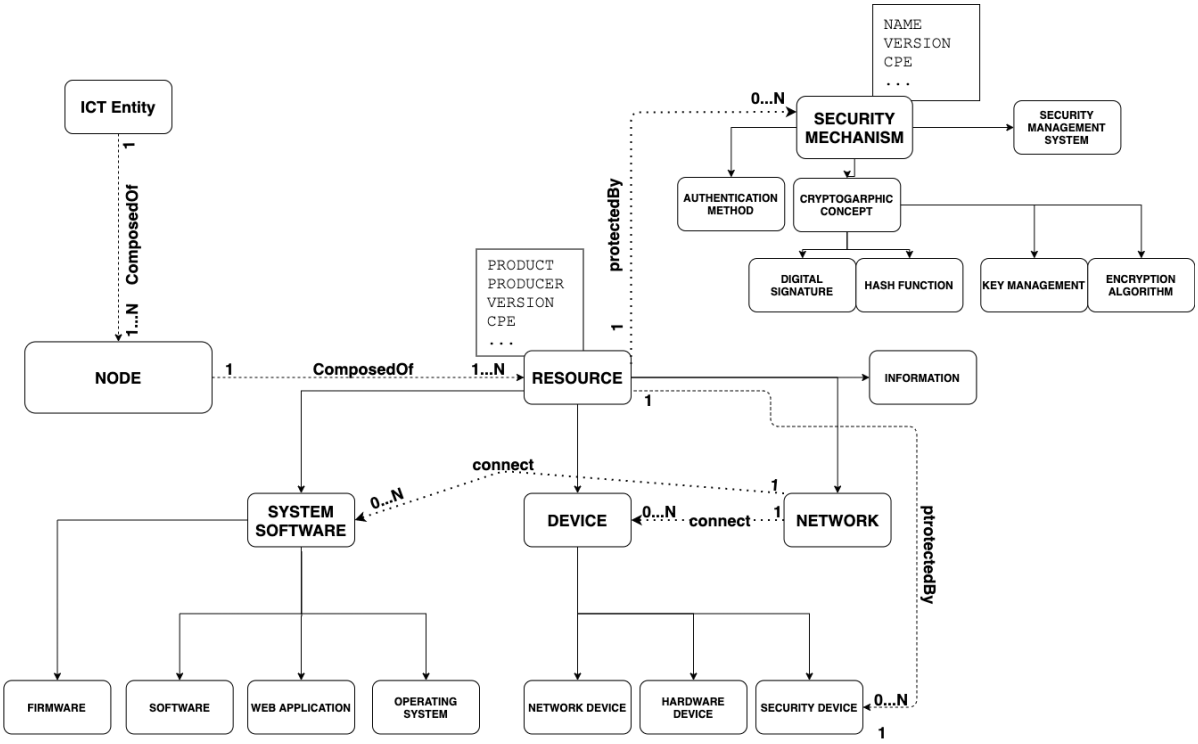


Fig. 5.5 Ontology for Modeling ICT Infrastructures

Figure 5.5 depicts the metamodel used for modeling ICT infrastructures. Particular attention has been placed on the the representation and organization of assets. As defined in [127], an asset can be either something valuable for the organization or everything that may contain vulnerabilities that can be menaced by threats. Hence, not only the stored data, the networks, and the digital services must be modeled but, more in general, every device and software connected to the ICT infrastructure that could be the target and/or the vector of possible attacks.

As briefly anticipated in Chapter 2 part of the motivations that drove this work stem from the Italian National Cybersecurity Agency's (ACN) creation of an ontology to support entities operating in the National Cybersecurity Perimeter (PSNC) in creating and maintaining the list of assets of their infrastructure. Some concepts, particularly the concept of *External Service*, were inherited from that ontology. While, in order to be compliant with well-known system modeling standards and to provide a more comprehensible and sharable metamodel, some high-level concepts related to the target ICT infrastructure have been taken from the TOGAF ArchiMate [126] enterprise architecture modeling standard, designed and developed by the Open Group[1]. In particular, referring to the technology layer described in the standard, the

[1] https://www.opengroup.org

concepts of (i) Node, (ii) Resource, (iii) System Software, (iv) Device, and (v) Network have been adopted.

Another key point in the design of the ICT infrastructure metamodel is the tight relation between system components and threats. Infrastructure components are organized according to the type of threats they can be targeted by and vice versa, as will be presented in Section 5.2.5 and Section 5.4. The sub-ontology is, thus, organized as detailed in the sequel of this section. For each class, we provide a brief description, its sub-entities and relationships, as well as why they have been introduced and what they represent.

The main class, ICTEntity, represents the infrastructure target of the analysis, which is related to class Node and by the composedOf relationship. The cardinality defined for the relationship means that if an ITCEntity individual is entered then Node individual must also necessarily be entered, but to provide a better modelization and ITCEntity individual can be composed of more than one Node individual.

The Node class is used to model a node of the infrastructure; as defined in TOGAF ArchiMate [126] a node represents a computational or physical resource that hosts, manipulates, or interacts with other computational or physical resources. Basically, the Node class is used as an aggregator of resources and can be seen as a collection of technology components of hardware and software that provide a specific service within the infrastructure; it can represent either a physical entity (i.e., a data center) or a digital service. Although not directly used for vulnerability management, threat modeling or risk evaluation, this class has been introduced to allow a better structural organization of resources belonging to different parts of the infrastructure. It is related to the class Resource by the composedOf relationship, the cardinality of the relationship here ensures that a Node individual is associated with at least one Resource individual. Moreover, to allow a better modeling approach, a Node is related to itself using the relationship composedOf to represent the fact that a Node can be composed of different nodes of the infrastructure. For this relation, the cardinality does not impose any restriction and a Node individual can be composed of zero to many other nodes.

The Resource class is used to model the assets of the infrastructure; as defined in TOGAF ArchiMate [126] a resource represents an asset owned or controlled by an individual or organization. It includes four main subclasses: (i) Network, (ii) System Software, (iii) Device, (iv)Information. These subclasses (and their subclasses as well) specialize the types of resources or assets the infrastructure may contain and have been introduced to provide both a more formal modeling, avoiding possible misunderstanding in the modeling phase, and a way to better map specific threats to assets in order to prevent the inclusion of threats that are not meaningful within the analyzed context. The Resource class is related to the classes Security

`Device` and `Security Mechanism` using the relationship `protectedBy`; the cardinality of the relationship expresses the fact that a resource can be protected by more than one security mechanism or device but also that no protection is implemented.

The `Resource` class is internally characterized by additional defined properties allowing a better description of the actual resource individual represented, which are inherently inherited by the `Resource` subclasses:

- Product: name of the product
- Producer: name of the producer
- Version: version number of the product
- CPE: Common Platform Enumeration: a naming scheme for IT systems, software, and packages identification, maintained by NIST[2].

All this information is needed to identify all the components of the target infrastructure and is required for the vulnerability management, as presented in Section 5.3.

The `Network` class is used to model a network of the infrastructure; as defined in TOGAF ArchiMate [126] a network represents the communication service interconnecting resources of the system. The `Network` class is related to other subclasses of `Resource` by the `connect` relationship, the cardinality of the relationship expresses the fact that a network can connect any number of devices or system software.

The `System Software` class, as defined in TOGAF ArchiMate [126], represents software deployed within the system that provides or contributes to a service. The `System Software` class is further divided into four subclasses: (i) `Firmware`, which represents software that provides low-level control for a device's specific hardware; (ii)`Software`; which represents everything that does not belong to other software classes, (iii)`Web Application`, which represents web applications; (iv)`Operating System`, which represents operating systems.

The `Device` class, as defined in TOGAF ArchiMate [126], represents physical IT resources upon which system software and artifacts may be stored or deployed for execution. The `Device` class is further divided into three subclasses: (i) `Network Device`, which represents a device that provides network functionalities (i.e., a router); (ii) `Hardware Device`, which represents a generic hardware device that does not fall under the other classes of devices (i.e., personal computer); (iii) `Security Device`, which represents a hardware device providing security-related functionalities (i.e., firewall, crypto-wallet, and so on).

[2] https://nvd.nist.gov/products/cpe

The class Information represents the critical information managed by the infrastructure. It describes data at rest, such as databases of user data.

The SecurityMechanism class is used to model security solutions used to protect the specific infrastructure component or resource. SecurityMechanism class is further specialized with three subclasses: (i) Authentication Method, which represents authentication protocols and mechanisms; (ii) Cryptographic Concept which represents cryptographic algorithms and protocols that can be used to protect resources of the infrastructure; (iii) Security Management System which represents security solutions such as Endpoint Detection Response (EDR) and monitoring systems.

The SecurityMechanism class is internally characterized by additional defined properties allowing a better description of the actual resource individual represented, which are inherently inherited by the SecurityMechanism subclasses:

- Name: name of the security mechanism
- Version: version number of the security mechanism
- CPE: of the security mechanism (if available).

5.2.2 Modeling Supply-chain

To better represent the analyzed infrastructure and also to consider, both in modeling and in security posture analysis, dependencies on external factors, the possibility of modeling the supply chain was included. The concepts included are, therefore, intended to allow one to indicate if a component of the infrastructure (be it a hardware or a software type resource) is provided by an external supplier, but also if the infrastructure (and consequently the organization) uses external services for its operations. In addition, it is possible to represent the fact that the infrastructure may provide an external service, which is important to be considered, also and especially at the security level. Figure 5.6 shows this part of the ontology.

The class ExternalService is used to model the supply chain and dependencies from external providers and suppliers. It is related to ICTEntity through two relationships: supply and dependsOn. The former represents something the infrastructure provides externally, such as a web service others may utilize, while the latter represents something the infrastructure depends on. The cardinality of the relationships expresses whether or not the infrastructure depends on or supplies one or more external services. Moreover, the class ExternalService is put in relation with the class Resource with the relationship isSuppliedBy to represent the

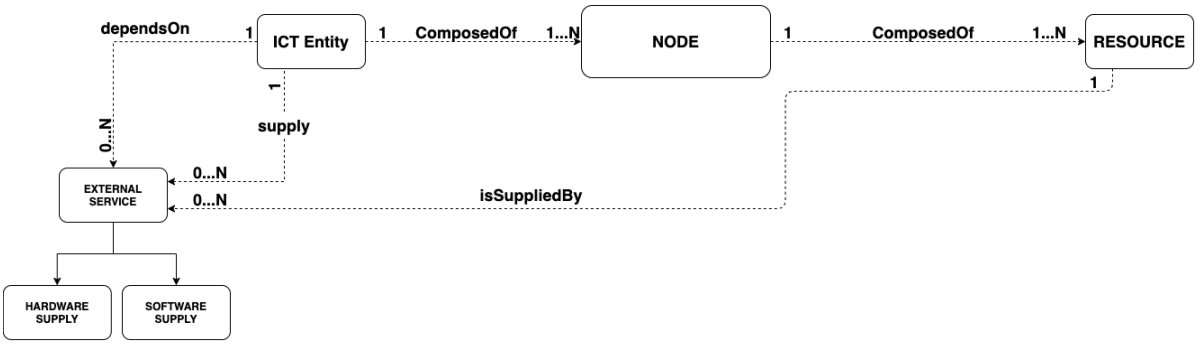


Fig. 5.6 Ontology for Modeling ICT Infrastructures Supply Chain

concept that a resource used and deployed within the infrastructure may come from an external supplier. Also in this case the cardinality of the relationship expresses the possibility or not of this happening.

`ExternalService` includes two subclasses: (i) `HardwareSupply`, (ii) `SoftwareSupply`. `HardwareSupply` represents the supply of hardware components, for example, outsourced production of devices. `SoftwareSupply` represents the supply of software components, such as outsourced development of software, or externally hosted digital services, such as web services or, in general, *something*-as-a-service.

5.2.3 Modeling Data Flow Diagram

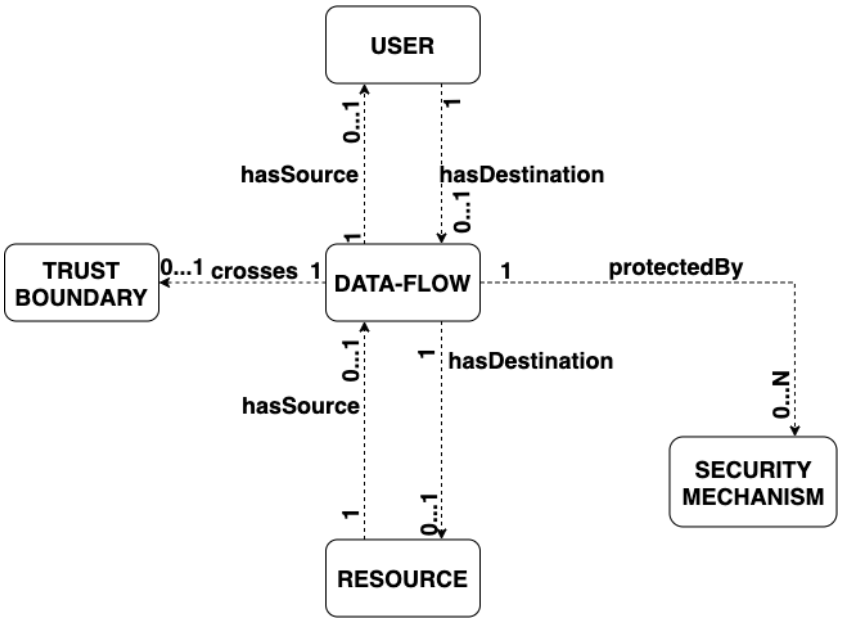


Fig. 5.7 Ontology for Modeling ICT Infrastructures Data Flow Diagram

Figure 5.7 depicts the *Data Flow* metamodel, the part of the ontology devoted to modeling the DFD[56], i.e., how components of the infrastructure communicate with each other.

The classes and relationships added to describe the data flow are quite simple. The main class is `Data Flow`; it represents a communication channel between the specified source and destination and is modeled by using two relationships: `hasSource` and `hasDestination`. The target of these two relationships can be a `User` class or a `Resource` class. The cardinality of these relationships describes the possibility of having as source or destination of a single data flow a maximum of one entity of type `User` or `Resource`.

The relationships `hasSource` and `hasDestination` are used also to describe the direction in which data are exchanged, if *software1* sends data to *software2* then the source of the data flow is *sfotware1* the destination is *software2*; this means that if there is the necessity of modeling an input-output interface for a given component than that component should be source and destination of two different individuals of type `Data Flow`.

Moreover, the `DataFlow` class can specify which security mechanism is in place for its protection using the relationship `isProtectedBy` with the entity `SecurityMechanism`. In this case, the relationship cardinality describes the possibility of having either zero, one or many security mechanisms in place for protecting a data flow.

The `TrustBoundary` class represents a change in the level of privileges between the source and destination of a data flow. This can be specified for a `DataFlow` class using the relationship `crosses`. In this case, the relationship cardinality expresses the possibility for a single data flow to cross a maximum of one trust boundary.

Finally, the `User` entity models the user interactions with the infrastructure.

5.2.4 Vulnerabilities

Figure 5.8 represents the part of the ontology devoted to vulnerability modeling. The goal here is to represent the vulnerabilities present in the analyzed system, represent their various characteristics, and tie them to the system components that are affected. In this way, it is possible to collect and represent the information needed for future analysis and security activities. The sub-ontology is, thus, organized as detailed in the sequel of this section. For each class, we provide a brief description, its sub-entities and relationships, as well as why they have been introduced and what they represent.

A vulnerability can be defined, according to MITRE as "*A flaw in a software, firmware, hardware, or service component resulting from a weakness that can be exploited, causing a*

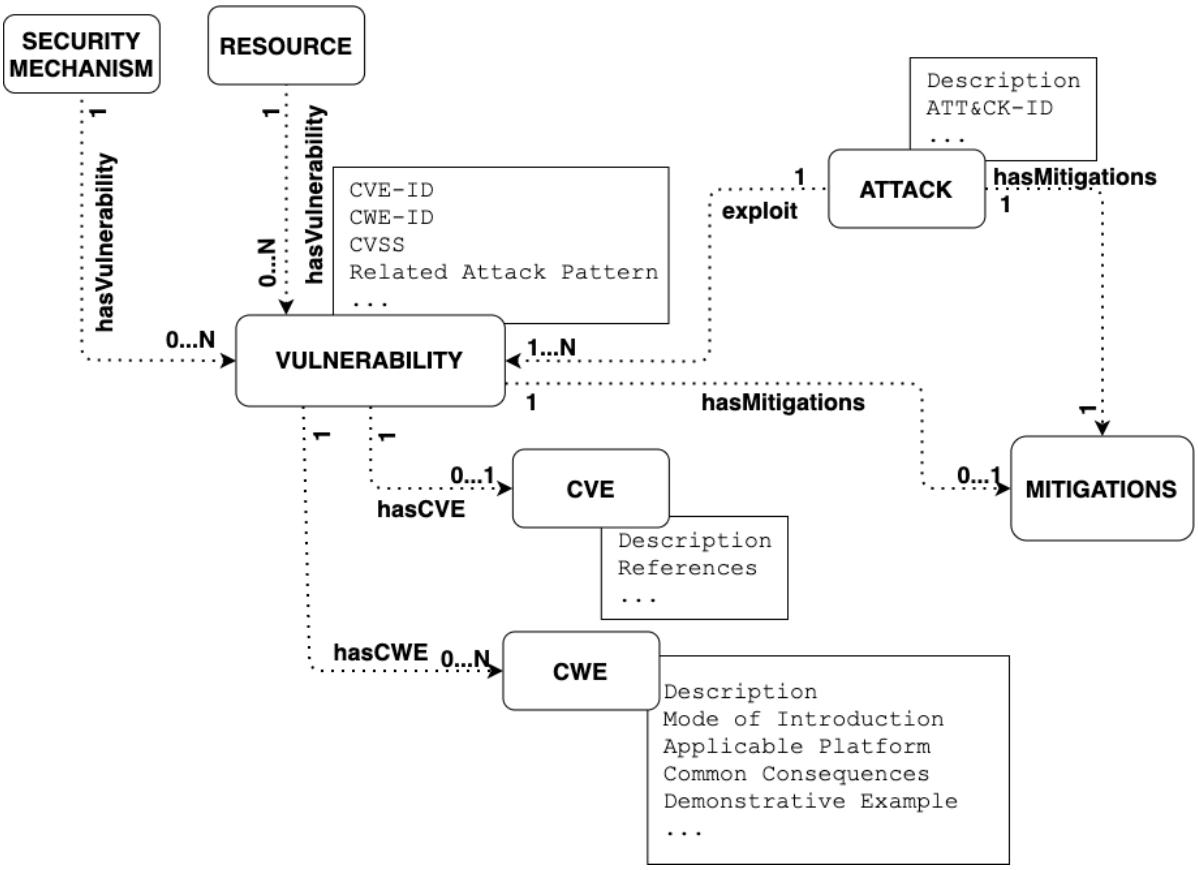


Fig. 5.8 Ontology for Modeling Vulnerabilities

negative impact to the confidentiality, integrity, or availability of an impacted component or components" [128]. Or, in other words, it is possible to define a vulnerability as a particular weakness, present in a specific component of a system, that can be *exploited* by an attacker to launch an *attack*. And it is based on this definition and the information that is collected in the various available vulnerability databases (i.e., CVE, CWE, and NVD) that the vulnerability ontology was created and designed.

The `Vulnerability` class is used to model the possible vulnerabilities affecting one or more resources of the system. It is related to classes `Resource` and `Security Mechanism` with the `hasVulnerability` relationship. Moreover, the `Vulnerability` class is related to other two classes `CVE` and `CWE` with, respectively, the relationship `hasCVE` and `hasCWE`; these two classes further collect information of the identified vulnerability from available sources. The cardinality of these two relationship describe the fact that each vulnerability is associated with one CVE entry, while each vulnerability may be associated with one or more weaknesses of the CWE database. Finally, the `Vulnerability` class is related to the `Mitigations` class with the relationship `hasMitigations` which represent the possible available mitigations for

the identified vulnerability. The cardinality of this relationship associate to one vulnerability the possible mitigations available. The relationships `hasCVE`, `hasCWE`, and `hasMitigations` cardinality allows also to relate the vulnerability to none of the connected classes, this possibility has been introduced to allow insertion of custom, discovered, vulnerabilities which does not have a link to available vulnerabilities databases.

The `Vulnerability` class is internally characterized by additionally defined properties that further enhance the description of the vulnerability:

- CVE-ID: used to store the related CVE-ID of the vulnerability.
- CWE-ID: used to store the related CWE-ID of the vulnerability.
- CVSS: used to store a possible evaluation of the vulnerability severity.
- Related Attack Pattern: contains a list of possible attack patterns exploiting the vulnerability.

The Common Vulnerability Scoring System (CVSS) [67] provide a standardized methodology for the severity evaluation of vulnerabilities; the standard is maintained by the FIRST organization[3] and it is currently at version 3.1. It is important to understand that CVSS is not a measure of risk, but rather a qualitative evaluation of the severity and impact of a vulnerability given the consequences it might have on a system, i.e, a vulnerability allowing remote code execution is ranked high or critical. The CVSS standard consists of three main metrics: *Base*, *Temporal*, and *Environmental*. Each of these is internally characterized by factors that are used to assign a quantitative value to the severity of the vulnerability. The *Base* metrics are the main ones and produce a value on a scale ranging from 0 to 10, while *Temporal* and *Environmental* metrics serve to better characterize, and thus modify the *Base* value, based on the context in which the vulnerability is assessed. In most cases, only the *Base* metric is used to rank vulnerabilities.

In its NVD database, NIST provides and maintains an evaluation of the *Base* metric, while dealing with incomplete information (i.e., vendor not disclosing certain details) by assigning to the vulnerability the maximum score possible. The CVSS scoring values are summarized in Table 5.1. From a vulnerability assessment operation perspective, this can be greatly beneficial to rank vulnerability by severity score and prioritize mitigation operations within the infrastructure.

The `CVE` class is internally characterized by additional defined properties allowing a description of the CVE database content, which mainly consists of a description of the vulnerability

[3] https://www.first.org/cvss/

Table 5.1 Base Metrics CVSS Scores

Severity	Base Score Range
None	0
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

and references to additional resources (i.e., vendor site of the vulnerable products or further information).

The `CWE` class is internally characterized by additional defined properties allowing a description of the CWE database content, these properties are:

- Description: provides a description of the weakness.
- Mode of introduction: provides a description of how the weakness can be introduced in the system.
- Applicable platform: provides additional information on whether the weakness is peculiar to a given technology or language.
- Common consequences: provides a list of the weakness common consequences in case of successful exploitation.
- Demonstrative Example: provides some example, usually in the form of code snippets of the weakness and possible attacks exploiting the weakness.

The `Attack` class has been introduced to represent possible attack patterns exploiting a given vulnerability. The `Attack` class is related to the `Vulnerability` class with the `exploit` relationship, while is related to the `Mitigations` class with the `hasMitigations` relationship.

The `Attack` class is internally characterized by additional defined properties allowing a description of the possible attack pattern exploiting the vulnerability and mapping with the ATT&CK database maintained by MITRE [96]. ATT&CK is a globally accessible Cyber Threat Intelligence (CTI) knowledge-base that collects adversary tactics, techniques and procedures (TTPs) based on real-world observations.

The `Mitigations` class, as already introduced, is used to model and collect possible available mitigations of identified vulnerabilities and attack patterns.

It is important to note that the information modeled in the vulnerability ontology, as will be presented in Section 5.3, assumes that the ontology is populated with known vulnerability information from the various knowledge-bases available (e.g., knowledge-bases from the MITRE and NVD), and thus some classes, properties, and relationships have been introduced precisely to model and represent the information contained and the links between these knowledge-bases. This, however, does not preclude that the ontology cannot be populated from custom information or from results obtained through VAPT campaigns carried out on the infrastructure under analysis; in this case, of course, some information may not be available (e.g., CVE and CWE IDs) or difficult to derive (e.g., links with CAPEC and ATT&CK knowledge-bases). The ontology is really intended to provide a means to try to standardize this kind of information so that it can be integrated and used easily.

5.2.5 Threats

The goal of this part of the ontology is to model the possible threats to which the infrastructure under analysis might be subjected. Threats are organized into categories based on the components of the infrastructure and the interactions that may occur between them; in fact, it was intended to design the threat sub-ontology in close relation to the ICT sub-ontology so that both the threats, but more importantly the *threat-component* mapping of the infrastructure, can be identified more accurately. This is a necessary operation for being able to support, as will be presented in Section 5.4, the threat modeling automation process. After that, the STRIDE class and the additional relationships with it were added to provide a simplified key to the identified threats and to support further analysis of the obtained results.

Figure 5.9 depicts the composition of the threat sub-ontology used for modeling threats. As defined by NIST [1] a Threats is "*Any circumstance or event with the potential to adversely impact organizational operations (including mission, functions, image, or reputation), organizational assets, individuals, other organizations, or the Nation through an information system via unauthorized access, destruction, disclosure, modification of information, and/or denial of service*".

The `Threat` class is used to model and organize threats, these are organized in different classes based on the type of infrastructure components they apply to and the interactions the components have with the surrounding environment. The `Threat` class is further divided into subclasses, the `Threat` class and its subclasses, that organize and specialize threats.

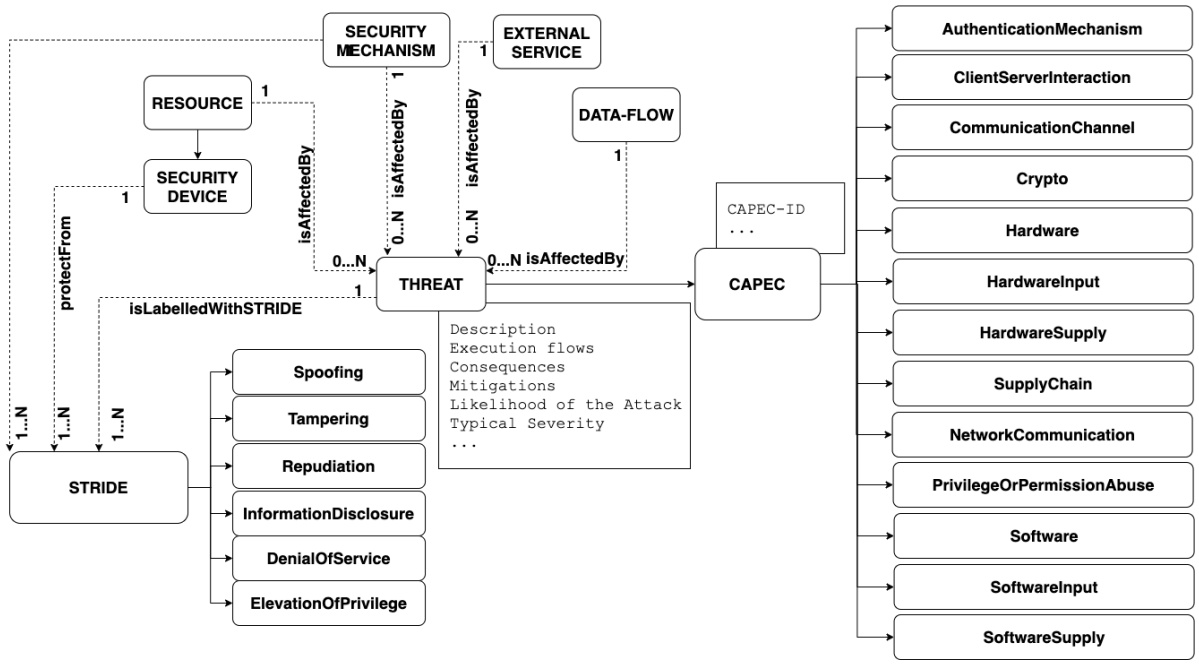


Fig. 5.9 Ontology for Modeling Threats

Currently, only the CAPEC database has been used as a source of information, hence the only direct subclass of `Threat` is `CAPEC`. The `Threat` class is related to the `STRIDE` class with the `isLabeledWithSTRIDE` relationship. The cardinality of the relationship represents the fact that a threat can affect one or more classes of STRIDE.

The `Threat` class is internally characterized by additional defined properties allowing a better description of the threat characteristics:

- **Description**: provides a description of the threat.
- **Execution flows**: provides some information about the possible way an attacker can perpetrate the attack.
- **Consequences**: provides a description of possible consequences in terms of confidentiality, integrity and so on.
- **Mitigations**: provides a list of possible mitigations that can be put in place to mitigate the threat.
- **Likelihood of the attack**: reflects the likelihood of attack success (threat materializing) on a scale {very low, low, medium, high, very high} in consideration of the attack prerequisites, targeted weaknesses, attack surface, skills, and resources required as well as effectiveness of likely implemented blocking solutions

- **Typical severity**: reflects the typical severity of the attack (consequences of the threat materializing) on a scale {very low, low, medium, high, very high}.

The source of information used for modeling threats, as well as populating the threat ontology, is the *MITRE CAPEC* knowledge-base. In particular, the abstraction level adopted is the *CAPEC* view *Domains of Attack*, which is composed of six categories: (i) **Software**, (ii) **Hardware**, (iii) **Communications**, (iv) **Supply Chain**, (v) **Physical Security**, and (vi) **Social Engineering**. Among those, the first four categories were selected to create the list of threats, while *Social Engineering* threats and *Physical Security* were not considered. The former was excluded because its effective application will require thoughtful modeling of human-related aspects, which are out of the scope of this work and will require further analysis. The latter was excluded for a similar reason; this category contains threats to physical security systems, such as locks and physical alarms, which are not taken into account in the proposed ICT infrastructure metamodel and are out of the scope of this work.

More specifically, CAPEC threats are organized following different levels of abstraction: each category contains *Meta Attack Patterns* which, in turn, contain *Standard Attack Patterns*. Each Standard Attack Pattern is composed of *Detailed Attack Patterns*. For sake of simplicity, the present paper focuses on the *Meta Attack Patterns* only, but the proposed reasoning and rules can be easily extended to the subclasses, by following the same described approach.

Starting from the *Domains of Attack* view, thirteen categories have been identified and the corresponding ontology classes created. CAPEC's classes have been associated with each entity by providing the corresponding ID, the description, and the applicable countermeasures. These are taken from the CAPEC knowledge-base, as well.

The `CAPEC` class is internally characterized by the property **CAPEC-ID** which refers to the corresponding entry in the CAPEC database; this property is inherently inherited by the `CAPEC` class subclasses. The `CAPEC` subclasses and threat categories identified are:

- Class `AuthenticationMechanism` refers to threats related to flaws in authentication systems;
- Class `ClientServerInteraction` refers to threats related to flaws in client-server communication;
- Class `CommunicationChannel` refers to threats affecting flaws in communication in general;
- Class `Crypto` refers to threats affecting flaws in applied cryptography;

- Class `Hardware` refers to threats affecting hardware devices;
- Class `HardwareInput` refers to threats affecting the hardware input/output system;
- Class `HardwareSupply` refers to threats affecting in particular the supply chain of hardware components.
- Class `SupplyChain` refers to threats affecting the supply chain, being them outsourcing production or supplying customers;
- Class `NetworkCommunication` refers to threats affecting network flaws in particular;
- Class `PrivilegeOrPermissionAbuse` refers to threats affecting flaws in the mechanism managing privilege levels;
- Class `Software` refers to threats affecting software;
- Class `SoftwareInput` refers to threats affecting software components' input interfaces;
- Class `SoftwareSupply` refers to threats affecting in particular the supply chain of software components.

Associating each threat with possible mitigations helps in providing a meaningful knowledge-base as a result of the threat modeling process. As described in Section 5.1, the results obtained can be used to better individuate remediation actions to be taken to improve the infrastructure security.

The classes `Resource`, `Security Mechanism`, `ExternalSerivice`, and `Data-Flow` are related to the class `Threat` with the `isAffectedBy` relationship. The relationship is used to associate the corresponding entities with the threats that have been identified. The cardinality of the relationship depicts the fact that a given entity may be affected by none, one or more threats.

The second part of the threat sub-ontology is devoted to model STRIDE. Simply, a `STRIDE` class was inserted with 6 sub-classes, each representing a STRIDE category. The `STRIDE` entity serves two purposes: (i) threats are associated with the STRIDE category they belong to, and (ii) security mechanisms and security devices adopted in the infrastructure are associated with the STRIDE category they mitigate. In order to associate security mechanisms and devices to the corresponding STRIDE category the relationship `protectFrom` is used. The cardinality of these relationships describes the fact that a security mechanism/device can mitigate one or more STRIDE categories.

Mapping between instances of the `CAPEC` class and the `STRIDE` sub-classes was performed by following the *CAPEC-STRIDE Mapping* project [129]. The project consists of binding the *Meta Attack Patterns* contained in CAPEC to the corresponding STRIDE category.

Table 5.2 summarizes the identified threat categories and the associated threats represented using the corresponding CAPEC-ID.

Table 5.2 Threat Categories and Associated Threats

Threat Category	**Associated CAPEC Threat IDs**
AuthenticationMechanism	21, 114, 115, 122, 560
ClientServerInteraction	22, 227
CommunicationChannel	117, 125, 130, 148, 151, 154, 161, 169, 192, 216, 224, 272, 594, 607, 94
Crypto	112
Hardware	154, 188, 212, 440, 607, 624
HardwareInput	74, 113, 116, 441
HardwareSupply	440
SupplyChain	116, 154, 176, 438, 439
NetworkCommunication	161
PrivilegeOrPermissionAbuse	233, 458, 554
Software	130, 154, 165, 175, 188, 212, 607
SoftwareInput	28, 74, 113, 116, 123, 137, 153, 173, 240, 242, 248, 441, 549
SoftwareSupply	184, 690

5.2.6 Risk

The last part of the ontology is the part added for modeling the information needed for risk assessment. The addition, in this case, is very simple, since most of the information needed has already been modeled in the rest of the ontology. In particular, for the risk part, the classes needed to model, and then enter when populating the ontology, the metrics and values needed for risk assessment were added. Figure 5.10 shows the part of the ontology responsible for modeling this information.

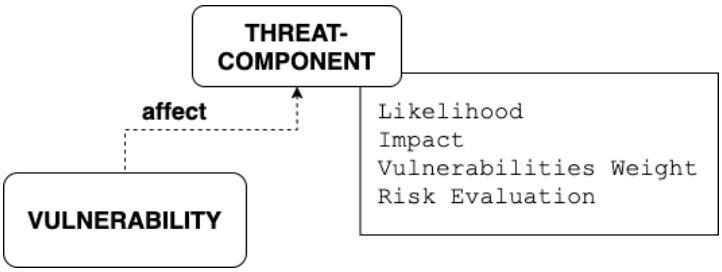


Fig. 5.10 Ontology for Modeling Threats

Taking the definition of cybersecurity risk provided by NIST presented in Chapter 3, risk can be defined as "*An effect of uncertainty on or within information and technology. Cybersecurity risks relate to the loss of confidentiality, integrity, or availability of information, data, or information (or control) systems and reflect the potential adverse impacts to organizational operations (i.e., mission, functions, image, or reputation) and assets, individuals, other organizations, and the Nation*"[4]. Thus linked to the concept of risk is that of threat, as the event that if it becomes real can cause damage to the infrastructure. It was therefore decided to link the information needed for risk assessment to that of threat.

In particular, the class `Threat-Component` has been added, which is used to collect information, derived from the threat modeling process (see Section 5.4), regarding which threats are associated with which component of the system. This association is used to be able to then subsequently calculate the measured risk for each identified threat. In addition, a relationship, `affect`, has been added that links the class `Vulnerability` with the class `Threat`. This relation was added to represent the link between a possible threat and the vulnerabilities present in the system that can be exploited by that threat, but, more importantly, also to represent the opposite link, thus the fact that vulnerabilities present within the system can contribute to the occurrence of a particular threat. Since the class `Threat-Component` represents a threat that afflicts a particular component, obviously in this case we refer only to the vulnerabilities identified for that particular component.

The class `Threat-Component` is internally characterized by four properties (i) `Likelihood`, (ii) `Impact`, (iii) `Vulnerabilities Weight`, (iv) `Risk Evaluation`:

- The `Likelihood` property represents the probability with which a threat may materialize and thus allows the value assessed for this quantity to be associated with a threat.

[4] https://csrc.nist.gov/glossary/term/cybersecurity_risk

- The `Impact` property represents the possible consequences, or damage, that the materialization of a threat may have on the infrastructure, and again the class allows a value to be associated with a threat that assesses this quantity.
- The `Vulnerability Weight` property represents the influence factor that vulnerabilities in the system may have in enabling, or facilitating, the materialization of a threat. This component was added, as will be presented later in Section 5.5, to allow the risk assessment operation to be performed and provide a result, from the ontology information, that is better contextualized with respect to the case under analysis.
- `Risk Evaluation` property represents the level of risk associated to the threat.

As will be shown in Section 5.5, all the above information items are currently retrieved from CAPEC given that the threats used in the threat modeling process are taken from CAPEC, but it is possible for a user to integrate or modify them at its own will.

5.3 Vulnerability Management: Proposed Approach

The goal here is to provide a mechanism that can be used by an organization to analyze and manage system vulnerabilities in a simplified manner. The proposed approach refers to the work presented in [113].

It should be noted that, in this case, reference is made to known vulnerabilities which can be found through available vulnerability databases. The use of an ontology to organize and collect the necessary information is of paramount importance in providing a standardized means of information representation and especially one that is machine-readable. In fact, as will be better detailed in this section the goal is to, starting from the information included within the `ICT sub-ontology` go on to identify in an automated way the known vulnerabilities of the various products used and collect the necessary information for effective management and in support of subsequent VAPT activities. Details and techniques used to perform penetration testing operations are out of the scope of this work, but the intention is to provide, thanks to the developed ontology, a tool to support these types of operations, which can, given a vulnerability of the system, provide guidelines on how the vulnerability can be exploited by combining both technical information, such as those present in CWE in the form of code snippets, and intelligence information that describes more broadly how an attack can be carried out, the techniques used and so on, referring to the information provided by MITRE ATT&CK. In

addition, the ontology also presents information on how a vulnerability can be mitigated, again leveraging primarily the two resources indicated above.

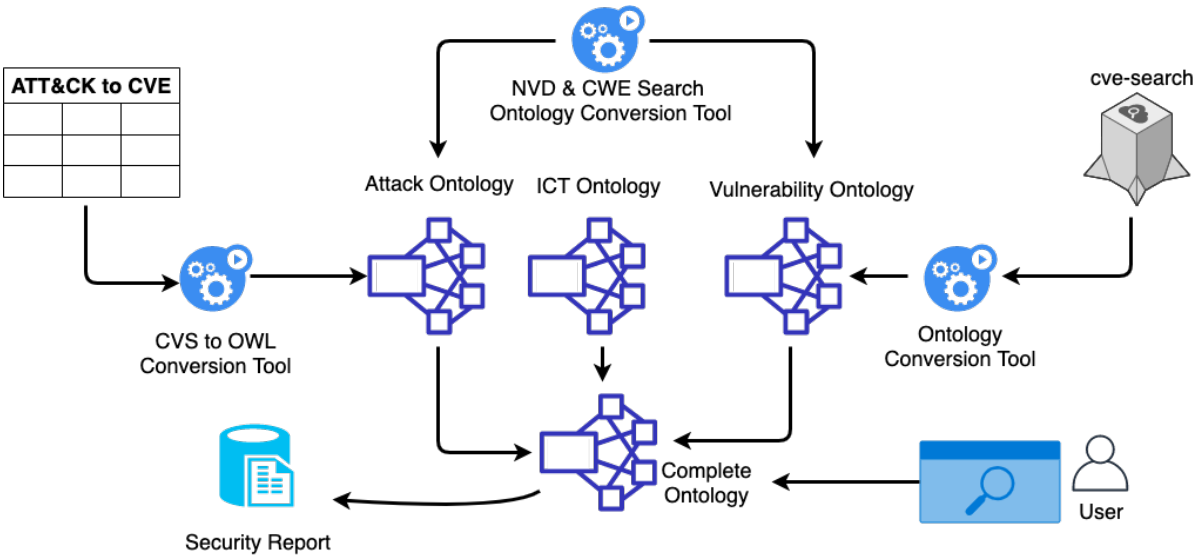


Fig. 5.11 Architecture of the Proposed Vulnerability Management Approach

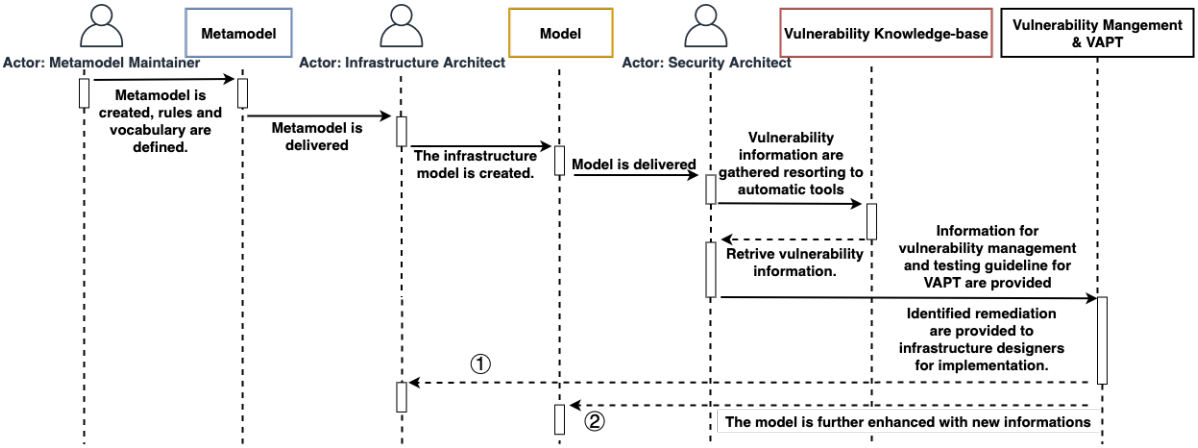


Fig. 5.12 Sequence Diagram of Activities Performed for Vulnerabilities Management

Figure 5.11 depicts a schema of the proposed approach while Figure 5.12 describes the order in which operations are performed.

Once the designed metamodel is ready, it is delivered to the *Infrastructure Architect* which populates the ontology with the target infrastructure information, thus generating the infrastructure model. As far as the description of the infrastructure under analysis is concerned, this must be done manually to include the highest possible level of detail.

Once the infrastructure model is ready, it can be populated with vulnerabilities information; as far as information on vulnerabilities, attacks and countermeasures are concerned, the process is made automatic thanks to available tools and ad hoc developed solutions. This information is retrieved from official vulnerabilities knowledge-bases (CVE, CWE, ATT&CK, and NVD) and inserted accordingly into the ontology model.

For greater clarity, it is possible to distinguish two main phases that are carried out chronologically one after the other: first, the part of the ontology related to the system vulnerabilities is populated. The information entered within the `ICT Ontology` is used to query the available vulnerability databases and the collected information is entered into the `Vulnerability Ontology`. After this step, identified the vulnerabilities, it is then possible to query the CWE and MITRE ATT&CK knowledge-bases to extract information regarding attacks and possible mitigation solutions available importing them within the `Attack Ontology`.

The population process of the `Vulnerability Ontology` can be summarized in the following steps:

- If the CPE value of a product is available, then resorting to NVD can identify known vulnerabilities and extract the corresponding `CVE-ID`, `CWE-ID` and, if available, `CVSS` scores.
- If the value of `CPE` is not available then the CVE database can be queried directly using the product/vendor name and its version as the search key. In this case, the more details are available on the product, the more the search can be targeted and less wide-ranging. Once the entries of interest have been identified, the corresponding `CVE-ID` can be extracted and the NVD database can be queried to extract `CWE-ID` and `CVSS` score if present.
- Once the CWE-ID is obtained, the MITRE database can be accessed to extract the additional information needed. This information is used to populate both the weaknesses part associated with the vulnerability: *description*, *mode of introduction*, *applicable platform*, *common consequences*, and *demonstrative example*. From the CWE database, information for the population of the class `Vulnerability` related to *relate attack patterns* can also be extracted, which contains a list of CAPEC database entries referenced and related to that weakness.

Once the CWE-ID is identified, information from the database can be retrieved in order to populate the corresponding ontology individual (see Section 5.2.4). From the CWE database are also extracted possible mitigation that can be adopted.

Querying and interacting with the CVE database can be made more efficient by using the "CVE-search"[5] tool, an open source project that consists of a local database that stores periodically synchronized information from CVE and a set of APIs to query the same. This allows faster queries and limits the sending of sensitive information over the internet. For the NVD database are provided APIs to either query directly the online database or download the content in a local database. It is required then to convert this information into OWL and add it to the ontology. This can be performed in several ways, an interesting project, M2Onto [130], describe a methodology to directly translate the content of a MongoDB[6] into an ontology providing the description of the ontological model to be filled; additionally are available Python libraries (e.g., Owlready2[7]) to work with ontologies that allow creation and modification of ontologies content.

Regarding the population of the ontology part of the attacks, the process can be summarized in the following steps:

- Once the CVE-IDs of the identified vulnerabilities have been obtained, it is possible to use an open source tool developed by MITRE, "attack_to_cve" [131], which allows one to extract, just from a given CVE-ID, the relevant ATT&CK knowledge-base entry and thus the `TechniqueID` related to the possible attack. This tool of MITRE, at April 2023, is still under development and is not complete: e.g., it is still not possible to find all the associations `CVE-ID-Technique-ID`.
- Alternatively, if the MITRE tool does not provide the desired results, it is possible to use the contents of the CWE database to identify possible attacks. In fact, once the information contained in `Related Attack Patterns`, namely a list of CAPEC entries, has been obtained, it is possible to trace back (again through the CAPEC database) to the relevant ATT&CK knowledge-base entries and thus obtain the reference `TechniqueID`.
- Finally, through the `TechniqueID` identified above, it is possible to query the MITRE ATT&CK knowledge-base to extract attack information as well the possible protection that can be used to mitigate the attack.

Once this process is finished the *Vulnerability Knowledge-base* is created, then the ontology with information about the identified known vulnerabilities. At this point the *Security Architect*, using this information can easily plan and implement vulnerability management and VAPT process. At this point, two generic scenarios are possible. ① represents the possibility of using

[5] https://www.cve-search.org/about/
[6] https://www.mongodb.com
[7] https://owlready2.readthedocs.io/en/latest/

the information identified in the ontology for vulnerability management, so vulnerabilities are prioritized, perhaps based on the value of *CVSS* identified, and the information about possible mitigation can be used and shared with the *Security Architect* infrastructure officer for implementation. ② represents the possibility of enhancing and refining the information in the ontology downstream of VAPT operations, perhaps starting precisely from the identified vulnerabilities and the guidelines that can be extracted to understand how the identified vulnerabilities might be exploited. In this case, the *Security Architect* can incorporate the results of a VAPT campaign to improve and enrich the content of the model, refining the content already present and adding the possible new vulnerabilities discovered in testing.

5.4 Threat Modeling: Proposed Approach

This section presents the proposed approach for automating the threat modeling process. Thus, the goal is to support analysts in the task of identifying possible threats afflicting a given infrastructure by attempting to automate it to the maximum possible extent. In particular, this section refers to the work proposed in [2], called **ThreMA**.

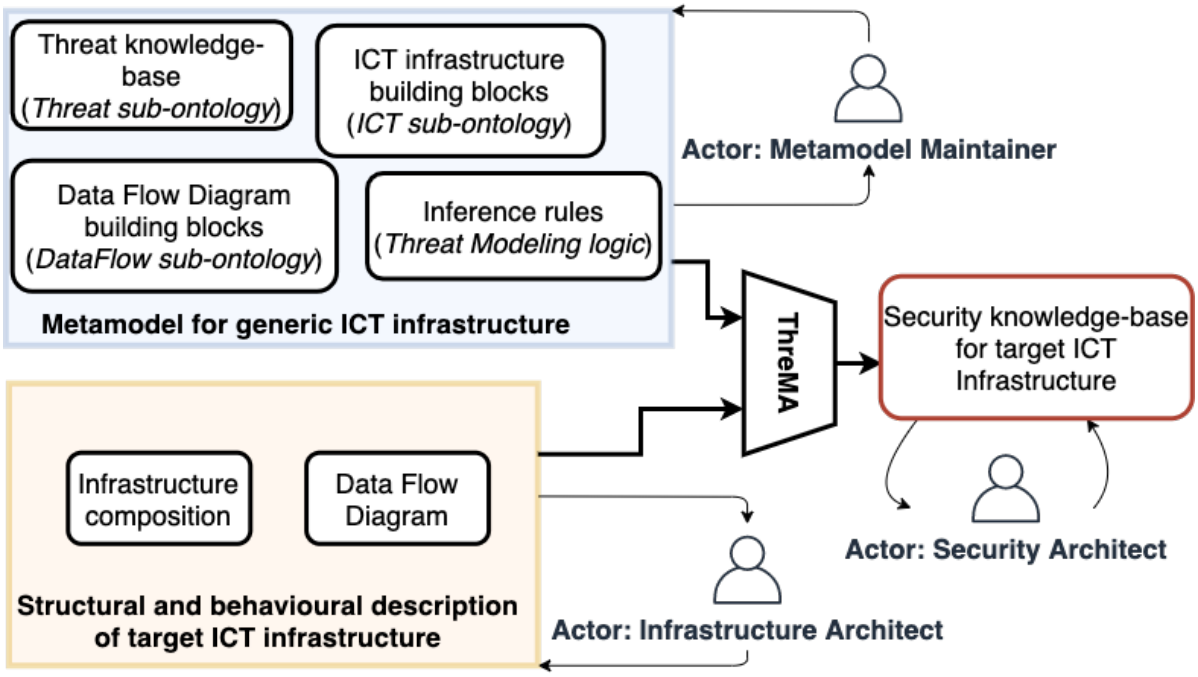


Fig. 5.13 Architecture of the Proposed Threat Modeling Approach (ThreMA) [2]

Figure 5.13 depicts the ThreMA architecture, showing the main components and actors involved. Two main inputs are required: the *ICT infrastructure metamodel* and the *structural*

and behavioural descriptions of the target ICT infrastructure. The ICT infrastructure metamodel (shortly, metamodel) is the ontology containing the set of formal modeling rules and vocabulary that allow for modeling ICT infrastructures and their threats. The threat knowledge-base provides the catalogue of threats organized by the type of infrastructure components they apply to, whereas rules, allow automatic reasoners to map the described infrastructure components to the affecting threats. The metamodel is maintained by the *Metamodel Maintainer*, i.e., the person or team in charge of developing and extending it.

The second input, on the other hand, represents the structural and behavioural descriptions, in terms of components and data flow, of the infrastructure under analysis. In this case, the *Infrastructure Architect*, i.e., the person responsible for the infrastructure under analysis, is in charge of designing, by exploiting the rules described in the metamodel, the target infrastructure, by creating a model containing assets, connections, and interactions among the various components.

On the basis of the provided inputs, ThreMA, by using the internal ontology reasoner, is able to automatically extract the threat model by applying the rules defined in the metamodel to the described infrastructure. The output of the process is a *security knowledge-base* that contains all the information needed by a *Security Architect* to identify critical points in the infrastructure, to plan the implementation and adoption of mitigation and remediation actions, and so on.

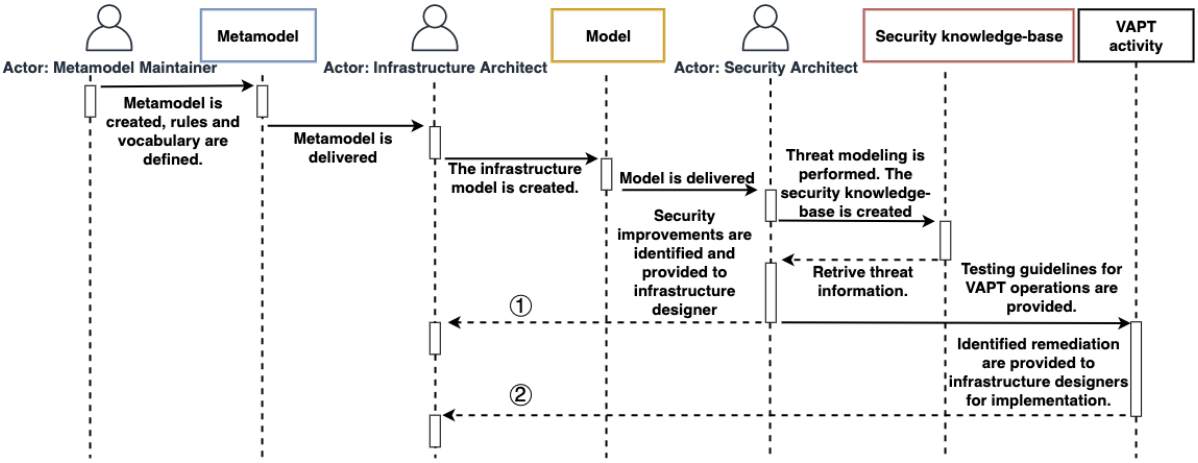


Fig. 5.14 Sequence Diagram of Activities Performed for Threat Modeling [2]

Figure 5.14 provides a sequence diagram showing the temporal sequence of steps performed by using the ThreMA tool. Everything starts with the creation of the metamodel (rules and vocabulary to model an ICT infrastructure) and of the threat knowledge-base. The metamodel is delivered to the *Infrastructure Architect* who designs the infrastructure under analysis. The

created *model* is then used to identify potential threats to the ICT infrastructure under analysis, through the ontology reasoner, by generating the relative *security knowledge-base*. The *Security Architect* can utilize the produced security knowledge-base either in the design phase or after the implementation.

Two use cases are shown in Figure 5.14 as well. ① represents threat modeling applied during the design phase; the *Security Architect* can use the *security knowledge-base* to identify security improvements and provide them to the *Infrastructure Architect* to modify the ICT infrastructure design. ② represents threat modeling applied to an ICT infrastructure already in operation. The *Security Architect* can utilize the gathered threat information to provide testing guidelines for VAPT (Vulnerability Assessment and Penetration Testing) activities or to identify vulnerable points in the ICT infrastructure and relative remediation plan to be provided to the *Infrastructure Architect* for implementation.

5.4.1 Threat Modeling Logic

This section presents the inference rules adopted for performing automatic threat modeling. Rules have been written using the SWRL language [117] and represent the logic behind threat modeling automation. Automatic reasoners can apply the defined rules to the ICT infrastructure model to map components to corresponding threats.

For each of the identified threat categories (see Section 5.2.5), rules have been provided and summarized in Table 5.3, using simplified pseudocode. The first column of the table contains the reference threat category; the second column describes the rules, the condition and the logic, determining when an infrastructure component is threatened by that category of threats.

Referring to Table 5.3, the first row describes the rules (second column) determining the conditions upon which threats in the category `AuthenticationMechanism` (first column) are applied. In detail, in the first rule presented the *antecedent* states that if exist an instance of the entity `Asset` which is connected to an instance of the entity `AuthenticationMechanism` by the relationship `isProtectedBy`, the *consequent* must hold, which state that the instance of the entity `Asset` is threatened by threats in the category `AuthenticationMechanism`. In the same way, all the reported rules should be read.

Is important to notice that the presence of security solutions, in the list of rules, does not remove the threats they mitigate. That was a deliberate choice, since problems related to improper use or implementation of security measures are often the source of attack vectors. Therefore, it was deemed more effective, in the threat modeling phase, to point out specific threats a security solution may introduce but not to remove threats it mitigates. In this way,

analysts are provided with a complete set of issues, useful to understand whether the security solutions adopted are effective. Anyway, the possibility to specify the STRIDE categories a security solution mitigates, and the mapping performed between threats and STRIDE, still provides information on which threats are being mitigated by which security solution.

Furthermore, it was decided not to base the threat modeling operation only on the identified vulnerabilities, this is to avoid having a partial and incomplete result. In fact, the vulnerabilities extracted in an automated way refer only to known vulnerabilities, and considering only vulnerabilities as threat factors would lead to disregarding those threats that could be introduced by vulnerabilities that are not yet known or by wrong implementation, integration, and interaction between infrastructure components.

Table 5.3 Inference Rules

Threat Category	**Rule(s)**
Authentication Mechanism	if `Resource isProtectedBy AuthenticationMethod`⇒`Resource` is threatened if `DataFlow crosses TrustBoundary` and `hasDestination Resource` ⇒ `Resource` is threatened
ClientServer Interaction	if `DataFlow hasSource Software(1)` and `hasDestination Software(2)` ⇒ `Software(2)` is threatened
Communication Channel	if `DataFlow` ⇒ `DataFlow` is threatened if `Network` ⇒ `Network` is threatened
Crypto	if `Resource isProtectedBy CryptographicConcept` ⇒ `CryptographicConcept` is threatened if `DataFlow isProtectedBy CryptographicConcept` ⇒ `CryptographicConcept` is threatened
Hardware	if `Device` ⇒ `Device` is threatened
Software Input	if `DataFlow hasDestination SystemSoftware` ⇒ `SystemSoftware` is threatened
Hardware Input	if `DataFlow hasDestination Device` ⇒ `Device` is threatened
Network Communication	if `Network` ⇒ `Network` is threatened
Privilege Or Permission Abuse	if `DataFlow crosses TrustBoundary` ⇒ `DataFlow` is threatened if `Firmware` ⇒ `Firmware` is threatened if `OperatingSystem` then `OperatingSystem` is threatened
Software	if `SystemSoftware` ⇒ `SystemSoftware` is threatened
Supply Chain	if `ICTEntity dependsOn ExternalService` ⇒ `ExternalService` is threatened if `ICTEntity supply ExternalService` ⇒ `ExternalService` is threatened if `Resource isSuppliedBy ExternalService` ⇒ `Resource` is threatened
Hardware Supply	if `Device isSuppliedBy HardwareSupply` ⇒ `Device` is threatened if `ICTEntity supply HardwareSupply` ⇒ `HardwareSupply` is threatened
Software Supply	if `SystemSoftware isSuppliedBy SoftwareSupply` ⇒ `SystemSoftware` is threatened if `ICTEntity supply SoftwareSupply` ⇒ `SoftwareSupply` is threatened

5.5 Risk Assessment: Proposed Approach

The goal in this case is to provide a tool that can support the risk assessment activity. Starting from the threats identified in the analyzed infrastructure, assign each of these threats a risk value, i.e., an assessment of the probability and consequences of it occurrence.

Figure 5.15 shows the architecture of the proposed solution. The risk assessment process is developed from the results obtained in the previous phases of vulnerability management (Section 5.3) and threat modeling (Section 5.4) with the goal of exploiting the information thus gathered to automate risk assessment.

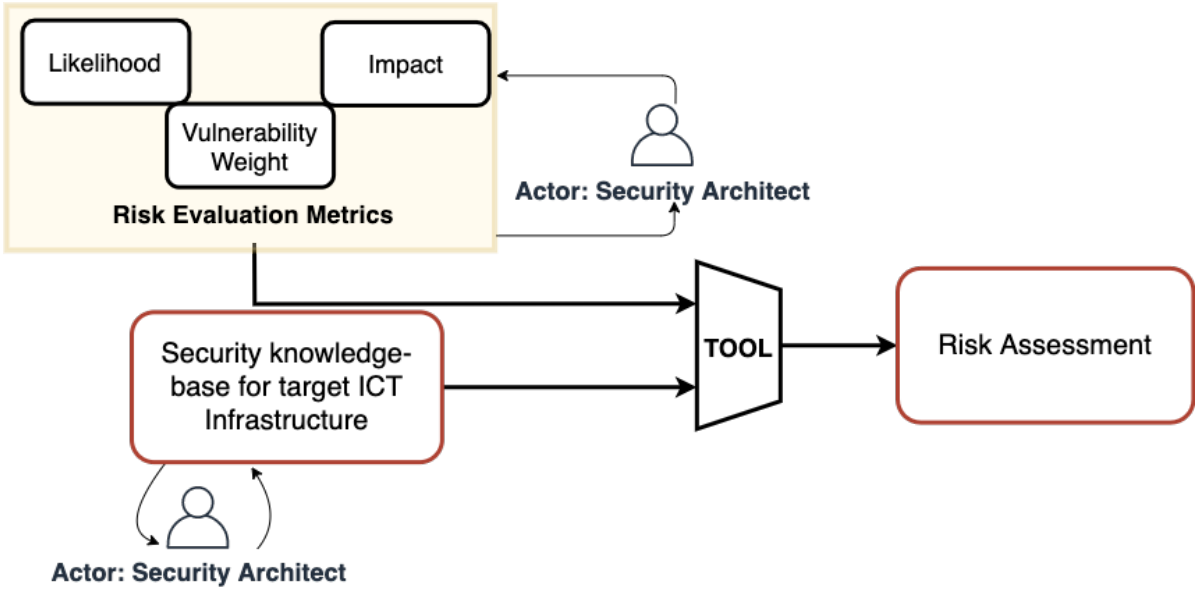


Fig. 5.15 Architecture of the Proposed Risk Assessment Approach

Reflecting what NIST defines in [1], the purpose of risk assessments is to inform decision makers and support risk responses by identifying (i) relevant threats to organizations; (ii) vulnerabilities; (iii) the impact (i.e., harm) to organizations that may occur given the potential for threats to exploit vulnerabilities; and (iv) the likelihood of harm occurring. In the specific case of this paper, of course, the risk assessment is not done at the level of the organization, but focusing specifically on the ICT infrastructure it deploys.

Thus, the first input needed for risk assessment is that represented by the *security knowledge-base*, the ontological model that collects all the security information items about the analyzed infrastructure obtained during the previous steps. These include the vulnerabilities identified for the infrastructure components, the composition of the infrastructure itself, and the threats identified for the infrastructure components.

The second input is the information needed to calculate the risk, and *risk evaluation metrics*, associated with each identified threat. The *likelihood*, hence the probability of occurrence of the given threat; the *impact*, hence the assessment of the degree of damage in case of materialization of the threat; and lastly the *vulnerability weight*, which represent a weight factor with respect to the presence of vulnerabilities affecting the given threat. These values are already present in the *security knowledge-base*, in fact they are extracted from the vulnerability databases and the CAPEC database used to represent the threats. It must be considered, however, that this information is context-unaware having, by necessity, to be generic, so there is still the possibility for the *Security Architect* to intervene and refine or modify these assessments to better consider the context in which the security assessment is being performed.

At this point, the information provided is used by the tool to calculate the risk associated with each identified threat. As far as risk calculation is concerned, it is not possible to use SWRL rules and ontology automatic reasoners, but one must rely on external tools developed ad-hoc. SWRL, similar to OWL, only supports monotonic inference [132]. Hence, given the SWRL monotonic nature, rules cannot be adopted to change already existing or add new information in an ontology. For this same reason, SWRL rules can not be used to remove information from an ontology. Moreover, given the nature of SWRL and ontology inference it is not possible to define rules in the form of `for each`, which for the proposed risk evaluation methodology are required.

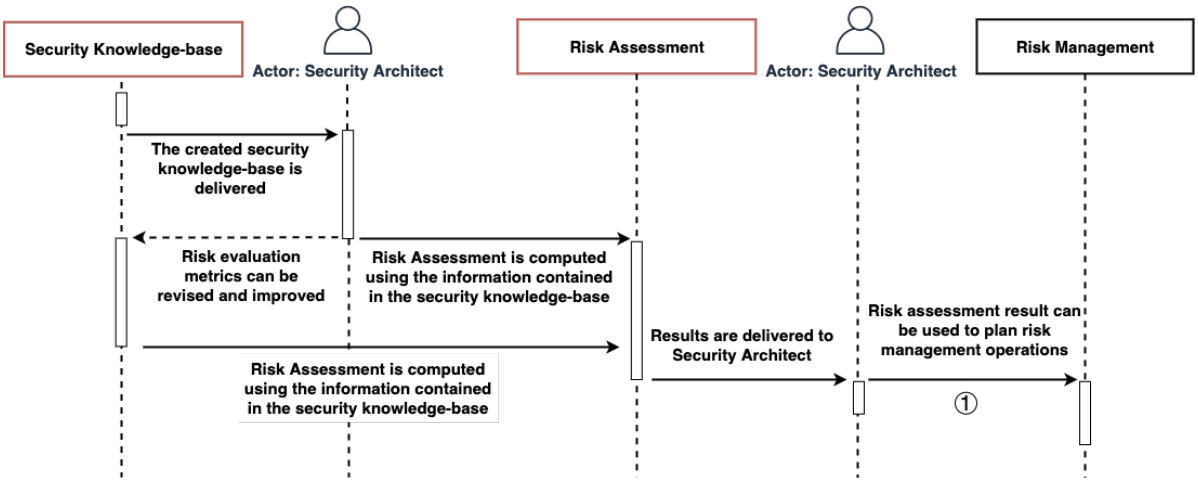


Fig. 5.16 Sequence Diagram of Activities Performed for Risk Assessment

A sequence diagram describing the time sequence of activities is shown in Figure 5.16.

As a first step, the *cybersecurity knowledge-base* is provided to the *Security Architect* who has the ability to add or refine the information needed to calculate the risk.

Then, using the provided tool, the Risk Assessment process is carried out in an automated manner and then a risk assessment is associated with each threat identified in the threat modeling phase.

The information resulting from the risk assessment is then fed back to the *Security Architect* who can then analyze it, prepare a ranking of the threats based on the associated risk, and so on.

① represents the ability on the part of the *Security Architect*, downstream of the risk assessment process, to use the information thus obtained, but also the information previously present in the *cybersecurity knowledge-base*, to prepare and plan risk management operations; such as the identification and mitigation of threats that pose a greater risk to the infrastructure and so on.

5.5.1 Risk Evaluation

There exist several approaches to risk analysis that differ with respect to the orientation or starting point of the risk assessment and the level of detail in the assessment. An analysis approach can be [1]:

- **Threat-oriented**: a threat-oriented approach starts with the identification of threat sources and threat events, and focuses on the development of threat scenarios; vulnerabilities are identified in the context of threats, and for adversarial threats, impacts are identified based on adversary intent.
- **Asset/impact-oriented**: an asset/impact-oriented approach starts with the identification of impacts or consequences of concern and critical assets, possibly using the results of a mission or business impact analyzes and identifying threat events that could lead to and/or threat sources that could seek those impacts or consequences.
- **Vulnerability-oriented**: a vulnerability-oriented approach starts with a set of predisposing conditions or exploitable weaknesses/deficiencies in organizational information systems or the environments in which the systems operate and identifies threat events that could exercise those vulnerabilities together with possible consequences of vulnerabilities being exercised.

Given the nature of the designed ontology as well as the type of information available and gathered during vulnerability identification and threat modeling, a *threat-oriented* approach was chosen to be used. In fact, as already indicated in Section 5.4, the threat modeling and identification process is performed independently of the identified vulnerabilities.

Since the identified vulnerabilities, at least referring to the automation process, refer only to known vulnerabilities of the system components and thus do not take into account vulnerabilities resulting from faulty integration of the components with each other or any custom components developed ad-hoc, using a *vulnerability-oriented* approach would have resulted in a potentially incomplete or misleading assessment. Whereas an *impact-oriented* approach, especially in threat identification, would have been more difficult to automate given the very context-dependent nature of the information being analyzed.

Given these considerations, then, in the risk assessment process we first identify the information resulting from the threat modeling process that identifies the threat-component pairing of the system. This information is saved in the entity of the defined `Threat-Component` ontology. In addition, also at this stage, vulnerabilities identified for the component itself that may impact the identified threat are associated with the *threat-component* pair. This association, which is made through the relationship `Vulnerability affect Threat-Component`, is made possible by the information collected related to the vulnerabilities. In fact, as indicated in Section 5.3 within the CWE database possible links between the weakness and the CAPEC database of attack patterns are identified, this information is then used to associate one or more vulnerabilities with the corresponding reference threat.

Considering that the threats currently considered in threat modeling refer only to the CAPEC category *Meta Attack Pattern*, the vulnerability-threat association is not always possible, since it is not necessarily the case that a given CWE is directly associated with a *Meta Attack Pattern*, but perhaps it is associated with a subcategory of the *Meta Attack Pattern*. To limit this problem, if the reference *Meta Attack Pattern* category cannot be directly identified, backward reasoning is done, starting from the CAPEC entry linked with the given CWE going back to the reference *Meta Attack Pattern*. This is possible because the CAPEC categories are linked by a parent-child relationship and it is therefore possible to trace back to the highest level category from a lower level.

After that, again, to the entity `Threat-Component` is added the information needed for risk assessment: (i) *Likelihood*, (ii) *Impact* and (iii) *Vulnerabilities Weight*. The first two are extracted with respect to the threat information in the *security knowledge-base*, in fact this information is available in the CAPEC database from where the threats used in threat modeling are extracted. These values are represented by MITRE on a scale {very low, low, medium, high, very high}, it is then necessary to transform them into numerical values according to Table 5.4. A problem here is that not for all the threats contained in CAPEC is it possible to find all the information (*Likelihood* and *Impact*) needed. In this case, a security analyst should intervene to make up for the lack by making an assessment with respect to the case under

analysis; however, this requires possessing the necessary skills for this type of assessment. In the proposed approach, a default value *high* is used when the information in question were not available.

Table 5.4 Risk Factor Conversion Scale

Very Low	**Low**	**Medium**	**High**	**Very High**
1	2	3	4	5

Regarding the quantity *Vulnerabilities Weight* this represents a risk weight factor associated with the threat calculated based on the characteristics of the vulnerabilities associated with that threat and identified in the system (CVSS score). This weight factor has been included to represent, also in the risk calculation phase, the fact that if there is a component in the system that is subject to a certain threat and that component has vulnerabilities that can be exploited to implement that given threat, and thus execute the corresponding attack, then that threat will have to have, prospectively, a higher level of risk given the presence of this type of predisposing condition in the system. The value associated with the vulnerability's CVSS score, found in the *cybersecurity knowledge-base* and extracted from the CWE database, is used to calculate the weight. In case of multiple vulnerabilities associated with the same threat, the highest CVSS value is taken.

Considering what has been presented so far and the information contained within the *cybersecurity knowledge-base* as a method for calculating risk, the following formula is provided:

$$R(T_i) = VW_i + \frac{L_i + I_i}{2} \tag{5.1}$$

where $R(T_i)$ is the computed risk value for the i-th threat, VW_i is the *vulnerabilities weight* associated with the i-th threat, L_i is the likelihood value of the i-th threat and I_i is the impact value of the i-th threat. Threat are identified as a *threat-component* pair; vulnerabilities affecting a particular threat are identified within the identified vulnerabilities of the component in the *threat-component* pair. The *Vulnerabilities Weight* represents, partially, the concept of *predisposing condition* which affect the overall likelihood of a threat or attack happening and contributes to identifying the overall risk for the given threat.

VW_i is calculated resorting to the following formula:

$$VW_i = \frac{CVSS(v)}{10}, \;\; v \in V_i \mid \forall CVSS(x) \in V_i \quad CVSS(v) \geq CVSS(x) \tag{5.2}$$

where, $CVSS(v)$ is the CVSS score of the vulnerability v, v is one of the vulnerabilities in the set V_i of vulnerabilities affecting the i-th threat and v is selected as the vulnerability with the greater CVSS score among vulnerabilities present in V_i.

Given this, the *Vulnerability Weight* of a threat can contribute to the increase of up to one level the risk evaluation based on *likelihood* and *impact* based on the most severe vulnerability affecting the given threat.

The formula returns a value between 0 and 6 which can be assigned to a qualitative scale {very low, low, medium, high, very high, critical} resorting to the conversion in Table 5.5. The calculated risk values then put the identified threats in perspective with each other, ranking them based also on the severity of identified vulnerabilities. This was done to make sure that, after the risk assessment process, during the risk management process a proper attention be placed first on those threats with vulnerabilities by potentially assigning them a higher ranking in the overall ranking with respect to threats that do not have affecting vulnerabilities.

Of course, everything has to be organized as a pipeline where, once dealt with the most important risks, the *cybersecurity knowledge-base* is updated and the overall process is repeated to have an updated and more precise threat ranking.

Table 5.5 Risk Evaluation Conversion Scale

0 - 1	**>1 - 2**	**>2 - 3**	**>3 - 4**	**>4 - 5**	**>5 - 6**
Very Low	Low	Medium	High	Very High	Critical

To summarize, the proposed method considers the fact that the risk assessment process is based only on the information contained in the open source and publicly available knowledge-bases trying to provide an effective assessment method. The resulting value of $R(T_i)$ can then be used to rank identified threats based on the risk evaluation to prioritize remediation action and plan cybersecurity management operations. Downstream of these operations, it is possible to revise the values and information entered into the *cybersecurity knowledge-base* to improve and make more precise the risk assessment process and enable the implementation of continuous risk management. In general, the proposed approach is only possible through the use of automation in the management of all the necessary information, in fact even considering a small or medium enterprise, the number of threats and vulnerabilities to be considered can easily explode. Moreover, the "component-threat-vulnerability association" was assumed as the point of view for risk calculation since we considered it the most comprehensive and detailed

one. Of course, different views and methods, which may vary depending on what information is considered versus what is contained in the *cybersecurity knowledge-base*, could be adopted.

Chapter 6

Results

This chapter presents some significant results obtained exploiting the proposed solution in some real cases. First, the cybersecurity knowledge-base creation phase consisting of the infrastructure model, vulnerability information, and threat identification (*threat modeling*) is be presented. Then the results related to the risk assessment of the identified threats are presented. Finally, some considerations with respect to the results obtained are outlined.

6.1 Preparing the *Cybersecurity Knowledge-base*

As a first operation, it is necessary to perform a *system modeling*, then model the target infrastructure to be analyzed through the metamodel defined through the ontology. Since all subsequent operations are based on the composition of the infrastructure this step is preliminary to subsequent operations. Figure 6.1 shows the infrastructure architecture used in the validation phase. The information represented refers to a real use case derived from a significant Italian public Stakeholder. The data reported have been appropriately anonymized, and any reference to the original infrastructure has been removed. The operation of *system modeling* must be performed manually.

Regarding the threat modeling phase, the approach (ThreMA), presented in Chapter 5 was used; this allows the threat modeling process to be automated thanks to the provided threat catalog (see Section 5.2.5) - that contains known threats affecting infrastructure components - and the inference rules (see Section 5.4) - that define the logic used in mapping threats to specific components.

Table 6.1 Threat Modeling Results

Infrastructure Component	**N. of Threats**	**Threat Categories**	**STRIDE**
Device	10	Hardware, HardwareInput	S T I
Operating System	28	Software, SoftwareInput AuthenticationMechanism PrivilegeOrPermissionAbuse	S D T I
Network	16	CommunicationChannel NetworkCommunication	S E D T I
Software	28	Software, SoftwareInput AuthenticationMechanism PrivilegeOrPermissionAbuse	S E D T I
Firmware	23	Software, SoftwareInput PrivilegeOrPermissionAbuse	S E D T I
PC3	18	Hardware, HardwareInput AuthenticationMechanism HardwareSupply, SupplyChain	S E D T I
Network devices	18	Hardware, HardwareInput Authentication Mechanism PrivilegeOrPermissionAbuse	S E D T I
Security devices	18	Hardware, HardwareInput Authentication Mechanism PrivilegeOrPermissionAbuse	S E D T I
DataFlow	15	CommunicationChannel	S E D T I
APP 1&9 APP 4&5	29	Software, SoftwareInput AuthenticationMechanism PrivilegeOrPermissionAbuse ClientServerInteraction	S E D T I
Storage	11	Hardware, HardwareInput, Crypto	S E D T I
External Service	7	SupplyChain, SoftwareSupply	T I
APP5, OS1	34	Software, SoftwareInput AuthenticationMechanism, SuplyChain PrivilegeOrPermissionAbuse, Software Supply	S E D T I
FIRM1	29	Software, SoftwareInput PrivilegeOrPermissionAbuse SupplyChain, SoftwareSupply	S E D T I

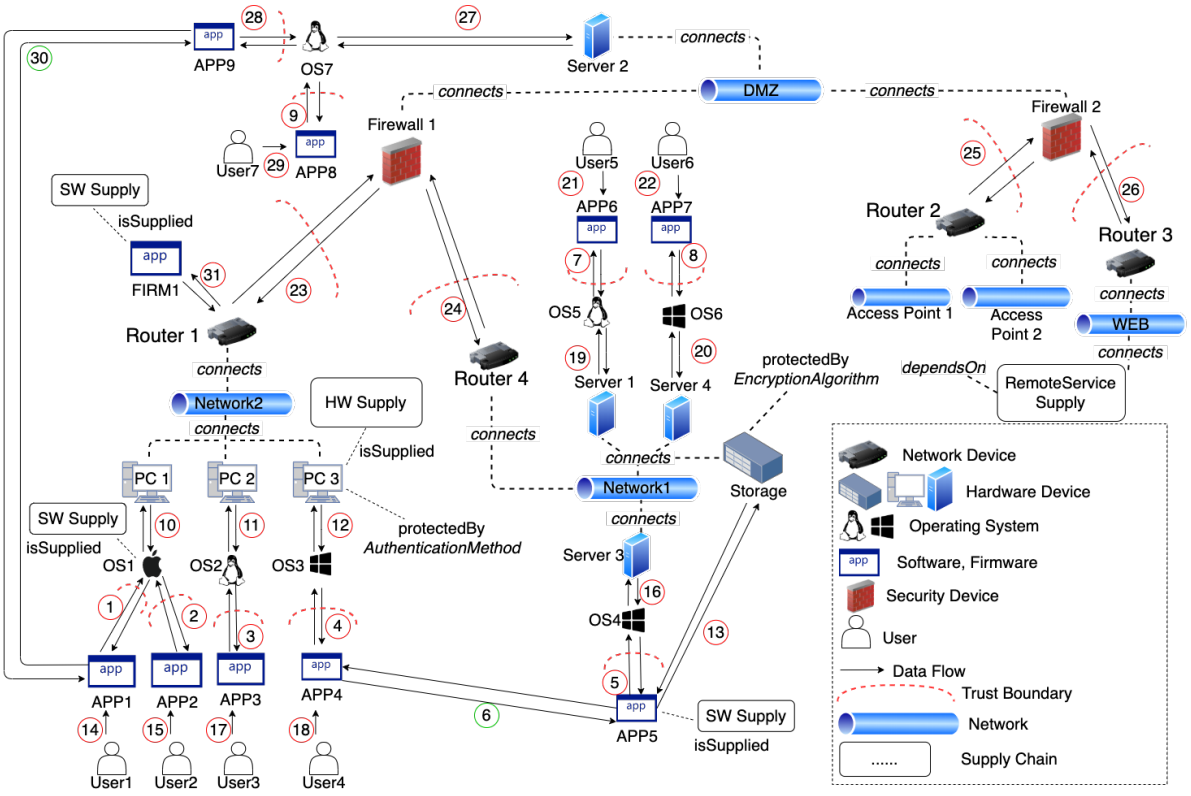


Fig. 6.1 Use-case Infrastructure

Following the sequence diagram presented in Section 5.1, once the construction of the metamodel was completed, the ontology was populated with information from the target infrastructure. Referring to Fig. 6.1, instances for the various hardware components, software, and networks were inserted and the data flow diagram was modeled. The legend in Fig. 6.1 shows which ontology entity is used to model the given infrastructure asset. Arrows define dataflow direction and, consequently, how it has been modeled with ontology entities and relationships (`DataFlow` `hasSource` and `hasDestination`). The created model consists of approximately 100 instances, each representing a component of the infrastructure.

Once the model was built, threat modeling was performed. The rules defined for the *Threat Modeling Logic* were applied resorting to the automatic reasoner Pellet [118]. The output represents the *Threat Model* of the infrastructure under analysis, and the process leads to the identification of more than 1,000 possible threats.

Table 6.1 summarizes the main and most significant results obtained; the first column describes the infrastructure components considered; the second column the number of threats identified; the third column the categories of the identified threats, and the last column the reference to STRIDE categories.

More in detail, all *hardware* individuals in the infrastructure are affected by threats in the categories `Hardware`, given the type of component, and `HardwareInput`, since they have an input interface. Moreover, *network devices* and *security devices* are also affected by threats in the classes `AuthenticationMechanism` and `PrivilegeOrPermissionAbuse` categories being targets of a dataflow crossing a trust boundary. Instances such as *PC3* and *Storage* (both hardware devices), being protected respectively by an authentication mechanism and an encryption mechanism, are also affected by threats of the corresponding categories. In addition, *PC3* is also an example of supplied hardware, having the relationship `isSuppliedBy` `HardwareSupply`, hence it is also affected by threats in categories `SupplyChain` and `HardwareSupply`

All *software* individuals in the infrastructure are affected by threats in categories `Software`, given the type of asset, and `SoftwareInput`, since they implement an input interface. Moreover, *software* individuals are also affected by threats in `AuthenticationMechanism` and `PrivilegeOrPermissionAbuse` categories because targets of a dataflow crossing a trust boundary. Instances of `Software` connected to data flow 6 and 30 (*APP1 - APP9* and *APP4 - APP5*) are example of what falls in the `ClientServerInteraction` category and are then associated with threats in the corresponding category. In addition, *APP5* is an example of supplied software, hence it is associated with threats in `SupplyChain` and `SoftwareSupply` categories.

All *operating system* and *firmware* individuals in the infrastructure are affected by threats in categories `Software`, given the type of asset, and `SoftwareInput`, since they implement an input interface. Moreover, they are also affected by threats in `AuthenticationMechanism` and `PrivilegeOrPermissionAbuse`. In addition, *OS1* and *FIRM1* are two examples of supplied software, hence they are associated with threats in the `SupplyChain` and `SoftwareSupply` categories.

All *dataflow* individuals are affected by threats belonging to the `CommunicationChannel` category.

All *network* individuals, on top of threats in the `CommunicationChannel` category, are also affected by `Network` threats.

Finally, the *Remote Service Supply*, modeling the use of externally supplied services, is associated with threats in the `SupplyChain` and `SosftwareSupply` category.

Each threat in the catalog is associated with possible countermeasures that can be adopted to mitigate attacks.

The complete model, the ontology containing both system and threat modeling, constitutes part of the *security knowledge-base* for the given ICT infrastructure. The generated knowledge

base can be queried by security analysts to extract useful information regarding the threats affecting the system and possible countermeasures. The mapping between the identified threats and the corresponding STRIDE categories provides an overall summary of the identified issues, while the association between threats and countermeasures allows easier identification of remediation actions to be adopted and/or implemented.

Always starting from the *system modeling*, it is possible to extract the known vulnerabilities of the various components. We can refer to the following significant examples. Consider that, again referring to the diagram in Figure 6.1, *OS1* is macOS 10.15 Catalina and that it refers to the following CPE *cpe:2.3:o:apple:mac_os_x:10.15.7:security_update_2020:*:*:*:*:*:*:*:*; while *APP5* is an on-premise installation of GitLab 13.10 that references the following CPE *cpe:2.3:a:gitlab:gitlab:13.10.2::::community:*:*:**; and finally that *FIRM1* is the firmware of a D-Link router with the following CPE *cpe:2.3:h:d-link:dir-825:r1:*:*:*:*:*:*:*:*.

At this point, by extracting the vulnerability information we get the result shown in Table 6.2.

Table 6.2 Number or extracted vulnerabilities for given products

Infrastructure component	**N. of Vulnerabilities**
OS1	1,220 matching entries in NVD
APP5	236 matching entry in NVD
FIRM1	1 matching entry in NVD

This simple example is also explicative of a known problem in literature regarding inconsistency of vulnerability databases information [133]. The content of public vulnerabilities databases is not always precise and can contain both false positive and false negative, hence reducing the effectiveness of using this information for security analysis. Solutions are currently being researched to improve the consistency of results, but solving this problem was out of the scope of this work.

Table 6.3 Example of extracted vulnerabilities information for Component OS1

CVE-ID	**CWE-ID**	**CAPEC-ID**	**Technique-ID**
CVE-2022-26691	CWE-269	CAPEC-122, CAPEC-233	T1548
CVE-2022-22582	CWE-59	CAPEC-154, CAPEC-122 CAPEC-165, CAPEC-153	1548, 1036.003
CVE-2022-26775	CWE-190	CAPEC-153	N/A
CVE-2022-26770	CWE-125	CAPEC-123	N/A
CVE-2022-26704	CWE-59	CAPEC-154, CAPEC-122 CAPEC-165, CAPEC-153	1548, 1036.003
CVE-2022-26698	CWE-125	CAPEC-123	N/A
CVE-2022-26697	CWE-125	CAPEC-123	N/A
CVE-2022-26688	CWE-59	CAPEC-154, CAPEC-122 CAPEC-165, CAPEC-153	1548, 1036.003
CVE-2022-22674	CWE-125	CAPEC-123	N/A

Table 6.4 Example of extracted vulnerabilities information for component APP5

CVE-ID	**CWE-ID**	**CAPEC-ID**	**Technique-ID**
CVE-2023-1708	CWE-77	CAPEC-248	N/A
CVE-2023-1072	CWE-400	CAPEC-227	N/A
CVE-2022-3513	CWE-79	CAPEC-242	N/A
CVE-2023-0050	CWE-79	CAPEC-242	N/A
CVE-2022-3381	CWE-601	CAPEC-137	N/A
CVE-2022-4138	CWE-352	CAPEC-212, CAPEC-116 CAPEC-21	1134, 1528 1539
CVE-2022-3411	CWE-20	CAPEC-123, CAPEC-175 CAPEC-233, CAPEC-248 CAPEC-153, CAPEC-22 CAPEC-137, CAPEC-242 CAPEC-130, CAPEC-165 CAPEC-28, CAPEC-115	1548, 1499.003 1036.003
CVE-2022-4201	CWE-918	CAPEC-115	1548
CVE-2022-3572	CWE-79	CAPEC-242	N/A

Table 6.5 Example of extracted vulnerabilities information for component FIRM1

CVE-ID	CWE-ID	CAPEC-ID	Technique-ID
CVE-2020-29557	CWE-119	CAPEC-123	N/A

Considering a subset of vulnerabilities, selecting them from the most recently added entries in the NVD database, it is possible to identify the links between CVE, CWE, CAPEC and ATT&CK required to populate the vulnerability metamodel. Tables 6.3, 6.4, and 6.5 summarize results for the vulnerability extraction of the component *OS1*, *APP5*, and *FIRM1*, respectively. The Tables report those vulnerabilities that contain the required information for identifying all the links (CVE to CWE to CAPEC). However, such an information is not always present in public vulnerabilities databases. The first column contains the CVE-ID of the vulnerability identified for that component, the second column contains the CWE-ID associated to the given vulnerability, the third column contains the CAPEC-IDs associated to the given vulnerability (i.e., which threat the vulnerability is affecting), while the last column contains, when available, the related Technique-ID in the ATT&CK knowledge-base.

The complete model, the ontology containing system modeling, threat modeling, and vulnerabilities constitutes the *security knowledge-base* for the given ICT infrastructure. The generated knowledge base can be queried by security analysts to extract useful information regarding threats and vulnerabilities affecting the system and possible countermeasures.

6.2 Performing Risk Assessment

Starting from the content of the *security knowledge-base* it is possible to perform the *risk assessment*. The association component-threat-vulnerability is performed following the approach described in Section 5.5 and the values necessary for the risk assessment are collected.

Figure 6.2 reports the information of the component *OS1*, highlighting those identified threat affected by an identified vulnerability. Vulnerability *CVSS* scores are retrieved from the NVD database, while threat *likelihood* and *impact* are retrieved from the CAPEC knowledge-base; missing information are indicated as *N/A*.

In turn, Figure 6.3 reports the information of the component *APP5*, highlighting those identified threat affected by an identified vulnerability. Vulnerability *CVSS* scores are retrieved from the NVD database, while threat *likelihood* and *impact* are retrieved from the CAPEC knowledge-base; missing information are indicated as *N/A*.

Threat-Component	Vulnerability Affecting Threat	Vulnerability CVSS	Threat Likelihood	Threat Impact
CAPEC-122	CVE-2022-26691	6.7	4	3
CAPEC-122	CVE-2022-22582	5.9	4	3
CAPEC-122	CVE-2022-26704	7.8	4	3
CAPEC-122	CVE-2022-26688	7.1	4	3
CAPEC-233	CVE-2022-26691	6.7	N/A	N/A
CAPEC-154	CVE-2022-22582	5.9	3	3
CAPEC-154	CVE-2022-26704	7.8	3	3
CAPEC-154	CVE-2022-26688	7.1	3	3
CAPEC-165	CVE-2022-22582	5.9	N/A	3
CAPEC-165	CVE-2022-26704	7.8	N/A	3
CAPEC-165	CVE-2022-26688	7.1	N/A	3
CAPEC-153	CVE-2022-22582	5.9	N/A	3
CAPEC-153	CVE-2022-26775	9.8	N/A	3
CAPEC-153	CVE-2022-26704	7.8	N/A	3
CAPEC-153	CVE-2022-26688	7.1	N/A	3
CAPEC-153	CVE-2022-22674	5.5	N/A	3
CAPEC-123	CVE-2022-26770	7.8	4	5
CAPEC-123	CVE-2022-26698	7.1	4	5
CAPEC-123	CVE-2022-26697	7.1	4	5

Fig. 6.2 Information for Risk Assessment identified for component *OS1*

Threat-Component	Vulnerability Affecting Threat	Vulnerability CVSS	Threat Likelihood	Threat Impact
CAPEC-122	CVE-2022-26691	6.7	4	3
CAPEC-122	CVE-2022-22582	5.9	4	3
CAPEC-122	CVE-2022-26704	7.8	4	3
CAPEC-122	CVE-2022-26688	7.1	4	3
CAPEC-233	CVE-2022-26691	6.7	N/A	N/A
CAPEC-154	CVE-2022-22582	5.9	3	3
CAPEC-154	CVE-2022-26704	7.8	3	3
CAPEC-154	CVE-2022-26688	7.1	3	3
CAPEC-165	CVE-2022-22582	5.9	N/A	3
CAPEC-165	CVE-2022-26704	7.8	N/A	3
CAPEC-165	CVE-2022-26688	7.1	N/A	3
CAPEC-153	CVE-2022-22582	5.9	N/A	3
CAPEC-153	CVE-2022-26775	9.8	N/A	3
CAPEC-153	CVE-2022-26704	7.8	N/A	3
CAPEC-153	CVE-2022-26688	7.1	N/A	3
CAPEC-153	CVE-2022-22674	5.5	N/A	3
CAPEC-123	CVE-2022-26770	7.8	4	5
CAPEC-123	CVE-2022-26698	7.1	4	5
CAPEC-123	CVE-2022-26697	7.1	4	5

Fig. 6.3 Information for Risk Assessment identified for component *APP5*

Figure 6.4 reports the information of the component *FIRM1*, highlighting those identified threat affected by an identified vulnerability. Vulnerability *CVSS* scores are retrieved from the NVD database, while threat *likelihood* and *impact* are retrieved from the CAPEC knowledge-base; missing information are indicated as *N/A*.

Threat-Component	Vulnerability Affecting Threat	Vulnerability CVSS	Threat Likelihood	Threat Impact
CAPEC-123	CVE-2020-29557	9.8	4	5

Fig. 6.4 Information for Risk Assessment identified for component *FIRM1*

Here another limitation of the proposed solution is brought to light; in fact, the necessary information is not always available (e.g., the evaluation of *likelihood* and *impact* for the given threat). By referring to open source and freely available information, this is to be taken into account. In this case, as explained in Section 5.5, the default values *4 - high* ware assigned where missing.

At this point, having collected all the necessary information, it is possible to proceed to the risk assessment by resorting to Formula 5.5.1. Figure 6.5 reports the result obtained considering the above-mentioned components. The first column reports the *threat-component* pair with associated values for *vulnerability weight*, *likelihood* and *impact*, and the second column reports the computed risk. Results for each *threat-component* pair are sorted from the highest to the lowest. The *vulnerability weight* selected for threats having multiple affecting vulnerabilities has been selected considering the most severe vulnerability (highest CVSS score) in the affecting vulnerability set.

Having computed the risk for each identified threat, it is possible to rank them based on the criticality level, allowing to use this information to address the most important threat in the infrastructure and to plan risk management activities accordingly with the help of the information stored in the *cybersecurity knowledge-base*.

A comparison of the possible results obtained with other commercial or open source tools is difficult to implement. The risk assessment process is based on the threats identified, and the threat modeling proposed in this Thesis work explicitly refers to the threats contained in the CAPEC database, while other tools use a different representation of threats, thus making a direct comparison of the gathered data practically impossible.

Threat {VW,Likelihood,Impact}	Risk
CAPEC-123-OS1 {0.78;4;5}	5,28
CAPEC-233-OS1 {0.67;4;4}	4,67
CAPEC-173-OS1 {0;4;5}	4,5
CAPEC-153-OS1 {0.98;4;3}	4,48
CAPEC-122-OS1 {0.78;4;3}	4,28
CAPEC-165-OS1 {0.78;4;3}	4,28
CAPEC-607-OS1 {0;4;4}	4
CAPEC-240-OS1 {0;4;4}	4
CAPEC-242-OS1 {0;4;4}	4
CAPEC-438-OS1 {0;4;4}	4
CAPEC-439-OS1 {0;4;4}	4
CAPEC-21-OS1 {0;4;4}	4
CAPEC-560-OS1 {0;4;4}	4
CAPEC-175-OS1 {0;3;5}	4
CAPEC-154-OS1 {0.78;3;3}	3,78
CAPEC-28-OS1 {0;4;3}	3,5
CAPEC-74-OS1 {0;3;4}	3,5
CAPEC-116-OS1 {0;4;3}	3,5
CAPEC-248-OS1 {0;3;4}	3,5
CAPEC-441-OS1 {0;3;4}	3,5
CAPEC-549-OS1 {0;3;4}	3,5
CAPEC-690-OS1 {0;3;4}	3,5
CAPEC-176-OS1 {0;4;3}	3,5
CAPEC-114-OS1 {0;4;3}	3,5
CAPEC-115-OS1 {0;4;3}	3,5
CAPEC-554-OS1 {0;3;4}	3,5
CAPEC-212-OS1 {0;3;3}	3
CAPEC-113-OS1 {0;3;3}	3
CAPEC-137-OS1 {0;3;3}	3
CAPEC-184-OS1 {0;4;2}	3
CAPEC-154-OS1 {0;3;3}	3
CAPEC-458-OS1 {0;2;4}	3
CAPEC-130-OS1 {0;3;3}	3
CAPEC-188-OS1 {0;2;2}	2

Threat {VW,Likelihood,Impact}	Risk
CAPEC-123-APP5{0.65;4;5}	5,15
CAPEC-21-APP5 {0.81;4;4}	4,81
CAPEC-22-APP5 {0.65;4;4}	4,65
CAPEC-233-APP5 {0.65;4;4}	4,65
CAPEC-242-APP5 {0.65;4;4}	4,65
CAPEC-175-APP5 {0.65;3;5}	4,65
CAPEC-227-APP5 {0,53;4;4}	4,53
CAPEC-173-APP5 {0;4;5}	4,5
CAPEC-248-APP5 {0,98;3;4}	4,48
CAPEC-116-APP5 {0.81;4;3}	4,31
CAPEC-28-APP5 {0.65;4;3}	4,15
CAPEC-153-APP5 {0.65;4;3}	4,15
CAPEC-115-APP5 {0.65;4;3}	4,15
CAPEC-165-APP5 {0.65;4;3}	4,15
CAPEC-607-APP5 {0;4;4}	4
CAPEC-240-APP5 {0;4;4}	4
CAPEC-438-APP5 {0;4;4}	4
CAPEC-439-APP5 {0;4;4}	4
CAPEC-560-APP5 {0;4;4}	4
CAPEC-212-APP5 {0.81;3;3}	3,81
CAPEC-137-APP5 {0.65;3;3}	3,65
CAPEC-130-APP5 {0.65;3;3}	3,65
CAPEC-122-APP5 {0;4;3}	3,5
CAPEC-74-APP5 {0;3;4}	3,5
CAPEC-441-APP5 {0;3;4}	3,5
CAPEC-549-APP5 {0;3;4}	3,5
CAPEC-690-APP5 {0;3;4}	3,5
CAPEC-176-APP5 {0;4;3}	3,5
CAPEC-114-APP5 {0;4;3}	3,5
CAPEC-554-APP5 {0;3;4}	3,5
CAPEC-154-APP5 {0;3;3}	3
CAPEC-113-APP5 {0;3;3}	3
CAPEC-184-APP5 {0;4;2}	3
CAPEC-154-APP5 {0;3;3}	3
CAPEC-458-APP5 {0;2;4}	3
CAPEC-188-APP5 {0;2;2}	2

Threat {VW,Likelihood,Impact}	Risk
CAPEC-123-FIRM1{0.98;4;5}	5,48
CAPEC-173-FIRM1 {0;4;5}	4,5
CAPEC-233-FIRM1 {0;4;4}	4
CAPEC-607-FIRM1 {0;4;4}	4
CAPEC-240-FIRM1 {0;4;4}	4
CAPEC-242-FIRM1 {0;4;4}	4
CAPEC-438-FIRM1 {0;4;4}	4
CAPEC-439-FIRM1 {0;4;4}	4
CAPEC-175-FIRM1 {0;3;5}	4
CAPEC-248-FIRM1 {0;3;4}	3,5
CAPEC-28-FIRM1 {0;4;3}	3,5
CAPEC-74-FIRM1 {0;3;4}	3,5
CAPEC-116-FIRM1 {0;4;3}	3,5
CAPEC-153-FIRM1 {0;4;3}	3,5
CAPEC-441-FIRM1 {0;3;4}	3,5
CAPEC-549-FIRM1 {0;3;4}	3,5
CAPEC-690-FIRM1 {0;3;4}	3,5
CAPEC-176-FIRM1 {0;4;3}	3,5
CAPEC-554-FIRM1 {0;3;4}	3,5
CAPEC-165-FIRM1 {0;4;3}	3,5
CAPEC-154-FIRM1 {0;3;3}	3
CAPEC-212-FIRM1 {0;3;3}	3
CAPEC-113-FIRM1 {0;3;3}	3
CAPEC-137-FIRM1 {0;3;3}	3
CAPEC-184-FIRM1 {0;4;2}	3
CAPEC-154-FIRM1 {0;3;3}	3
CAPEC-458-FIRM1 {0;2;4}	3
CAPEC-130-FIRM1 {0;3;3}	3
CAPEC-188-FIRM1 {0;2;2}	2

Legenda

Very Low	Low	Medium	High	Very High	Critical

Fig. 6.5 Example of computed risk for *threat-component* pairs

6.2.1 Considerations

Some considerations related to the proposed solution. As already mentioned in various parts of the Thesis, the proposed solution has limitations that in general can be identified in the use of open source security knowledge-bases. In fact, these are not always complete with all the necessary information [133] and this can lead to a final evaluation that is not completely correct [134], a factor that can limit the effectiveness of security solutions leveraging those data. These sometimes present inconsistencies in the information they contain, especially when considering the information linking these databases together. In fact, it is not always possible, starting from a given vulnerability (CVE-ID), to trace the corresponding weaknesses (WE-ID) and threats (CAPEC-ID) or attack techniques used (Technique-ID). This results, therefore, in the inability in these situations to fully exploit the information potentially available to arrive at (in the specific case of the work proposed here) a risk assessment. This could potentially be remedied through a huge amount of manual work, aimed at analyzing the information contained in these databases to find its links and synergies. This well known problem is recently beginning to be addressed by exploiting Machine Learning for the analysis and correlation of this large amount of information: examples can be identified in the work of Kanakogi et al. [135] trying to fill the gaps between CVE and CAPEC, linking vulnerabilities with the corresponding attack pattern, or of He et al. [133], proposing a mechanism to derive the missing information from the NVD database through the analysis of similar vulnerabilities that possess this information.

Regarding the vulnerability identification part, in the proposed solution, known vulnerabilities are identified from infrastructure components, leveraging external knowledge from vulnerability databases. This, potentially, provides only a partial view on the problem, since unknown vulnerabilities, or vulnerabilities of components developed within the organization, are not identified and it is not possible to do so. In general, referring to these data as sources of information results in the of lack a more precise contextualization of the results obtained. Overcoming these problems generally requires targeted VAPT activities to identify vulnerabilities that are not known a priori. The use of an ontology for data representation, however, leaves room for the user to enrich knowledge-base information with VAPT results, ensuring interoperability with the rest of the information in the ontology.

The same reasoning applies for threats. While threat modeling automation is a positive achievement of this work and certainly contributes to lessening the complexity of security evaluation activities for a given infrastructure, leveraging on publicly available knowledge for threat evaluation can provide a less precise result of the assessment. The knowledge available in CAPEC regarding threats evaluation is generic and to obtain a more precise result threats

should be evaluated considering the context in which the assessment activity is carried out (See Section 3.1).

Another limitation of the proposed solution is the lack of contextualization with respect to the risk assessments made. Obviously, public databases must be generic in the information they present, abstracting and detaching themselves from individual cases. This could lead in some cases to assessments that are not necessarily wrong, but perhaps to the identification of over- or under-estimated levels of risk with respect to the case under analysis. In this case it becomes necessary to try to incorporate data into the proposed model that will help improve the contextualization of the information to improve the accuracy of the final assessment [112].

Given the goal of this work to automate the security management process to the maximum extent, the result achieved is nevertheless satisfactory. Moreover, the proposed solution can be used, in general, as a means to collect and organize security information related to VAPT and threat assessment activities and then combine this information in an automated way to obtain a complete view of the risk on the infrastructure under analysis and through which to plan an effective risk management process.

Chapter 7

Conclusions

The present Thesis presented the results obtained during the research path pursued over the past few years. The original goal was to develop a system that could support users and security analysts in the activity of assessing the security posture of an ICT infrastructure and collect the information needed to plan risk management and VAPT activities.

A foundational component of the presented solution is the ontology around which everything else has been built. The ontology in fact provides the vocabulary necessary for the representation of all the necessary information items, to define its relationships and their mutual links. The first challenge faced was to define a representative metamodel of a generic ICT infrastructure, through which it would have been possible for a user to model the system to be analyzed. Particular care was taken in defining concepts and relationships to enable precise modeling and in sufficiently detailing the description of the system, both semantically and at the level of components and composition. In fact, this is a foundational aspect to allow proper identification of the necessary security information. The second point addressed was information about security vulnerabilities in the system. The metamodel for the ICT infrastructure has thus been enriched by adding the concepts and relationships necessary to identify which vulnerabilities are present in the system, how these are placed in relation to the components of the system, and to collect and represent all that information necessary (e.g., attack patterns through which vulnerabilities can be exploited, details regarding the weaknesses that make up the identified vulnerabilities, and so on) to a more in-depth analysis (e.g., through VAPT operations). In parallel, in the ontology metamodel, we defined and designed a part dedicated to the representation of threats and how they could affect the system (*threat modeling*). Finally, the concepts necessary for the risk assessment of identified threats (*risk assessment*) were added.

An automation process was then designed on the defined ontology, with the goal of automating to a maximum extent the fundamental parts of the *risk assessment* process, namely *vulnerability assessment* and *threat modeling*. To support the *vulnerability assessment* process, a *vulnerability management* approach was proposed that, from the information about the infrastructure under analysis, collected through the designed ontology metamodel, was able to identify known vulnerabilities of the infrastructure components from open source and publicly accessible knowledge bases (i.e, NVD, CVE, CWE) and to identify and extract all the information needed to detail these vulnerabilities by referring to CTI knowledge bases (CAPEC and ATT&CK). From the combination of these sources of information, it is possible to extract vulnerability mitigation guidelines, exploit examples, and so on. This is all information that is critical for guiding targeted *VAPT* activities.

To support the *threat modeling* process, a catalog of reference threats (extracted from the CAPEC knowledge-base) was created and the rules necessary for the complete automation of the threat identification process were defined. This process plays a key role in the security posture assessment phase. The process of *threat modeling* is entirely performed through the use of the designed ontology: for each component of the infrastructure, the logic driving the identification of its threats was developed leveraging inference rules and ontology reasoners.

Finally, having the vulnerability and threat information items available made it possible to properly combine them to perform the *risk assessment*. In this way, it is possible to provide a security analyst with a ranking of threats that places the focus on those that provide the greatest danger to the analyzed infrastructure and with the necessary information (i.e., mitigations) that can be leveraged to define an effective *risk management* process.

Certainly, the key contribution was related to the automation of the *threat modeling* process, a process that still sees mainly a purely manual execution, which finds several challenges in its automation.

The design of an ontology to support users in risk analysis activities goes in the direction of providing a tool to Italian public and private Stakeholders operating within the PSNC (see Chapter 2) that can help in the execution of security analysis activities required by the current Italian regulation. The proposed solution can be, in perspective, integrated with the work done by the Italian National Cybersecurity Agency (ACN) on the ontology for asset management, that they have developed, to provide a tool that allows a standardized execution of risk analysis by the Stakeholders involved and that allows for an easy sharing and comparison of the results obtained, as well as a means to assess the evolution and improvements implemented over time to mitigate the identified risks.

Considering the limitations identified in Chapter 6.2.1 surely some of the future developments will go toward improving the proposed solution to obviate the present limitations. Some guidance has already been provided, but the problem of information inconsistency within the MITRE and NVD databases will be addressed; recently some work has begun to address this issue by attempting to exploit machine learning, with promising results [133, 135], to identify or derive missing information and links between databases. Of particular importance for improving the work presented in this paper are the identification of the links between vulnerabilities (CVE), weaknesses (CWE), threats (CAPEC) and attack techniques (ATT&CK), but also the very assessments (likelihood and impact) of the threats contained in CAPEC, which are not always present, but are necessary and fundamental to be able to achieve the risk assessment.

The use of an ontology for information representation and thus a formal, machine-readable vocabulary, opens the possibility for a variety of applications. Certainly one of the issues that will be investigated in the future will be the possibility of using the defined ontology model as an input for the creation of a digital-twin within a cyber range [136] (isolated testbed for VAPT activities, training and so on). This would allow the possibility of easily having a realistic replica of the analyzed system on which to do further analysis and/or cybersecurity activities. Moreover, the proposed solution is not without limitations, in fact, referring to open source knowledge-bases as a source of information certainly lacks contextualization with respect to the context in which the analysis activity is being carried out. Furthermore, the automation process inserts a certain degree of false positives (i.e., threats or vulnerabilities that after an analysis could be identified as not present or pertinent) in both vulnerability identification and threat identification.

Moreover, the extension of the ontology metamodel to also consider different technological domains (e.g., IoT, CPS, cloud and so on) will definitely be addressed in future developments, not only at the infrastructure modeling level, but also in the threat modeling part to identify threats peculiar to the added technological domains [137, 27, 85]. This will certainly allow a better applicability of the proposed solution in different contexts and with greater accuracy in the results obtained. An extension of the metamodel would also go in the direction of providing the necessary information for a better contextualization of the information [112] also through a better characterization of the processed data to understand on what type of data the threats/vulnerabilities are going to impact and thus consequently modify the risk assessment performed.

Other improvements that will be addressed will concern a more in-depth analysis of how a threat or vulnerability may propagate within the system, e.g., in a manner similar to what is

described in the Attack Tree methodology for threat modeling [63]. This can allow a better analysis of the consequences of the occurrence of a threat or what the exploit of a vulnerability may entail. An important source of information that can be used for this type of activity is ATT&CK from MITRE, but, again, there is a lack of explicit linkage (even of temporal succession) between the various techniques that can be used for an attack [138]. Finally, additional activities that can be undertaken in the future concern the presentation aspect of the results, so that the analysis performed can be more easily understood. For example, in order to reduce the amount of information presented to the user, an operation could be done to aggregate the threats identified for each component taking into consideration the resulting issue of risk aggregation [139].

Those are still open challenges in this field and will certainly be the fundamental focus around which to define the future steps of the research work presented in this Thesis.

References

[1] Guide for conducting risk assessments. Standard SP 800-30 Rev. 1, National Institute of Standards and Technology, Gaithersburg, USA, 2012.

[2] Fabio De Rosa, Nicolò Maunero, Paolo Prinetto, Federico Talentino, and Martina Trussoni. Threma: Ontology-based automated threat modeling for ict infrastructures. *IEEE Access*, 10:116514–116526, 2022.

[3] Fortune. There's a huge surge in hackers holding data for ransom, and experts want everyone to take these steps. Available at https://fortune.com/2022/02/17/ransomware-attacks-surge-2021-report/, 2021. Accessed: 2023-04-10.

[4] Steve Morgan. Cybercrime To Cost The World $10.5 Trillion Annually By 2025. Available at https://cybersecurityventures.com/hackerpocalypse-cybercrime-report-2016/, 2020. Accessed: 2023-04-10.

[5] Kaspersky. What is WannaCry ransomware? Available at https://www.kaspersky.com/resource-center/threats/ransomware-wannacry. Accessed: 2023-04-10.

[6] Cybersecurity & Infrastructure Security Agency. Executive Order on Strengthening the Cybersecurity of Federal Networks and Critical Infrastructure. Available at https://www.cisa.gov/executive-order-strengthening-cybersecurity-federal-networks-and-critical-infrastructure, 2017. Accessed: 2023-04-10.

[7] ENISA. Executive Order on Strengthening the Cybersecurity of Federal Networks and Critical Infrastructure. Available at https://www.enisa.europa.eu/topics/cybersecurity-policy/nis-directive-new, 2023. Accessed: 2023-04-10.

[8] Atle Refsdal, Bjørnar Solhaug, and Ketil Stølen. Cyber-risk management. In *Cyber-risk management*, pages 33–47. Springer, 2015.

[9] Information technology - security techniques - information security risk management. Standard ISO/IEC 27005:2018, International Organization for Standardization, Geneva, CH, 2018.

[10] Risk management framework for information systems and organizations: A system life cycle approach for security and privacy. Standard SP 800-37 Rev. 2, National Institute of Standards and Technology, Gaithersburg, USA, 2018.

[11] Information security management systems - guidelines for information security risk management. Standard BS 7799-3:2017, British Standards Institution, London, UK, 2017.

[12] Interoperable eu risk management framework. Standard, European Union Agency for Cybersecurity, Attiki, GRC, 2022.

[13] Igor Linkov and Alexander Kott. Fundamental concepts of cyber resilience: Introduction and overview. *Cyber resilience of systems and networks*, pages 1–25, 2019.

[14] National Institute of Standard and Technology. Open Security Controls Assessment Language. Available at https://csrc.nist.gov/Projects/Open-Security-Controls-Assessment-Language, 2023. Accessed: 2023-04-10.

[15] Raúl Aranovich, Muting Wu, Dian Yu, Katya Katsy, Benyamin Ahmadnia, Matthew Bishop, Vladimir Filkov, and Kenji Sagae. Beyond nvd: Cybersecurity meets the semantic web. In *New Security Paradigms Workshop*, pages 59–69, 2021.

[16] Raúl Riesco and Víctor A Villagrá. Leveraging cyber threat intelligence for a dynamic risk framework: Automation by using a semantic reasoner and a new combination of standards (stix™, swrl and owl). *International Journal of Information Security*, 18(6):715–739, 2019.

[17] Brian E Ulicny, Jakub J Moskal, Mieczyslaw M Kokar, Keith Abe, and John Kei Smith. Inference and ontologies. *Cyber Defense and Situational Awareness*, pages 167–199, 2014.

[18] Santiago Moral-García, Santiago Moral-Rubio, David G Rosado, Eduardo B Fernández, and Eduardo Fernández-Medina. Enterprise security pattern: A new type of security pattern. *Security and Communication Networks*, 7(11):1670–1690, 2014.

[19] Jeffrey A Ingalsbe, Louis Kunimatsu, Tim Baeten, and Nancy R Mead. Threat modeling: diving into the deep end. *IEEE software*, 25(1):28–34, 2008.

[20] Stephan Aier, Sabine Buckl, Ulrik Franke, Bettina Gleichauf, Pontus Johnson, Per Närman, Christian M Schweda, and Johan Ullberg. A survival analysis of application life spans based on enterprise architecture models. *Enterprise modelling and information systems architectures*, 2009.

[21] MacDonnell Ulsch. *Cyber threat!: how to manage the growing risk of cyber attacks*. Wiley Online Library, 2014.

[22] Kym Bergmann. The growing cyber threat. *Asia-Pacific Defence Reporter (2002)*, 40(2):12–13, 2014.

[23] Maxime Frydman, Guifré Ruiz, Elisa Heymann, Eduardo César, and Barton P Miller. Automating risk analysis of software design models. *The Scientific World Journal*, 2014, 2014.

[24] Wenjun Xiong and Robert Lagerström. Threat modeling–a systematic literature review. *Computers & security*, 84:53–69, 2019.

[25] Valentina Casola, Alessandra De Benedictis, Massimiliano Rak, and Umberto Villano. Toward the automation of threat modeling and risk assessment in iot systems. *Internet of Things*, 7:100056, 2019.

[26] Ioannis Zografopoulos, Juan Ospina, Xiaorui Liu, and Charalambos Konstantinou. Cyber-physical energy systems security: Threat modeling, risk assessment, resources, metrics, and case studies. *IEEE Access*, 9:29775–29818, 2021.

[27] Salman Manzoor, Tsvetoslava Vateva-Gurova, Rubén Trapero, and Neeraj Suri. Threat modeling the cloud: an ontology based approach. In *International Workshop on Information and Operational Technology Security Systems*, pages 61–72. Springer, 2018.

[28] Abel Yeboah-Ofori and Shareeful Islam. Cyber security threat modeling for supply chain organizational environments. *Future internet*, 11(3):63, 2019.

[29] Andreas Ekelhart, Stefan Fenz, Markus D Klemen, and Edgar R Weippl. Security ontology: Simulating threats to corporate assets. In *International Conference on Information Systems Security*, pages 249–259. Springer, 2006.

[30] Margus Välja, Fredrik Heiding, Ulrik Franke, and Robert Lagerström. Automating threat modeling using an ontology framework. *Cybersecurity*, 3(1):1–20, 2020.

[31] MITRE. About the cve program. Available at https://www.cve.org/About/Overview.

[32] MITRE. About cwe. Available at https://cwe.mitre.org/about.

[33] National Institute of Standards and Technology. National vulnerability database. Available at https://nvd.nist.gov.

[34] MITRE. About the capec. Available at https://capec.mitre.org/about/index.html.

[35] D.L. 14 giugno 2021, n. 82. Disposizioni urgenti in materia di cybersicurezza, definizione dell'architettura nazionale di cybersicurezza e istituzione dell'agenzia per la cybersicurezza nazionale. https://www.gazzettaufficiale.it/eli/id/2021/06/14/21G00098/SG, June 2021.

[36] D.Lgs. 18 maggio 2018, n. 65 . Attuazione della direttiva (ue) 2016/1148 del parlamento europeo e del consiglio, del 6 luglio 2016, recante misure per un livello comune elevato di sicurezza delle reti e dei sistemi informativi nell'unione. https://www.gazzettaufficiale.it/eli/id/2018/06/09/18G00092/sg, May 2018.

[37] 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. Available at https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32016L1148, Jul 2016.

[38] D.L. 21 settembre 2019, n. 105 . Disposizioni urgenti in materia di perimetro di sicurezza nazionale cibernetica. https://www.gazzettaufficiale.it/eli/id/2019/09/21/19G00111/sg, September 2019.

[39] D.P.C.M. 15 giugno 2021. Individuazione delle categorie di beni, sistemi e servizi ict destinati ad essere impiegati nel perimetro di sicurezza nazionale cibernetica, in attuazione dell'articolo 1, comma 6, lettera a), del decreto-legge 21 settembre 2019, n. 105, convertito, con modificazioni, dalla legge 18 novembre 2019, n. 133. https://www.gazzettaufficiale.it/eli/id/2021/08/19/21A05087/sg, June 2021.

[40] D.P.R. 15 giugno 2021. Regolamento recante attuazione dell'articolo 1, comma 6, del decreto-legge 21 settembre 2019, n. 105, convertito, con modificazioni, dalla legge 18 novembre 2019, n. 133. https://www.gazzettaufficiale.it/eli/id/2021/04/23/21G00060/sg, June 2021.

[41] Andrea Lucariello Lorenzo Damiano, Luisa Franchina. Perimetro di sicurezza nazionale cibernetica, in gazzetta ufficiale tutti i beni e servizi inclusi. *cybersecurity360*, 08 2021.

[42] D.P.C.M. 14 aprile 2021, n. 81. Regolamento in materia di notifiche degli incidenti aventi impatto su reti, sistemi informativi e servizi informatici di cui all'articolo 1, comma 2, lettera b), del decreto-legge 21 settembre 2019, n. 105, convertito, con modificazioni, dalla legge 18 novembre 2019, n. 133, e di misure volte a garantire elevati livelli di sicurezza. https://www.gazzettaufficiale.it/eli/id/2021/06/11/21G00089/sg, April 2021.

[43] Framework nazionale per la cybersecurity e data protection. Standard, CIS Sapienza, Cyber Security National Laboratory of the National Interuniversity Consortium for Informatics, 2018.

[44] Eugene A Rosa. Metatheoretical foundations for post-normal risk. *Journal of risk research*, 1(1):15–44, 1998.

[45] Integrating cybersecurity and enterprise risk management (erm). Standard NISTIR 8286, National Institute of Standards and Technology, Gaithersburg, USA, 2020.

[46] Information technology — security techniques — information security management systems — overview and vocabulary. Standard ISO/IEC 27000:2018, International Organization for Standardization, Geneva, CH, 2018.

[47] Framework for improving critical infrastructure cybersecurity. Standard Version 1.1, National Institute of Standards and Technology, Gaithersburg, USA, 2018.

[48] Italian cybersecurity report - national cybersecurity framework. Standard, CIS Sapienza, Cyber Security National Laboratory of the National Interuniversity Consortium for Informatics, 2015.

[49] 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). Available at https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex:32016L1148, April 2016.

[50] ISACA. Cmmi levels of capability and performance. https://cmmiinstitute.com/learning/appraisals/levels. Accessed: 2023-03-20.

[51] Metodologia per il cybersecurity assessment con il framework nazionale per la cybersecurity e la data protection. Standard, CIS Sapienza, Cyber Security National Laboratory of the National Interuniversity Consortium for Informatics, 2021.

[52] Information security, cybersecurity and privacy protection — evaluation criteria for it security — part 1: Introduction and general model. Standard ISO/IEC 15408-1:2022, International Organization for Standardizqation, Geneva, Switzerland, 2022.

[53] European Commission. The EU Cybersecurity Act. Available at https://digital-strategy.ec.europa.eu/en/policies/cybersecurity-act, 2019. Accessed: 2023-04-10.

[54] Aaron Marback, Hyunsook Do, Ke He, Samuel Kondamarri, and Dianxiang Xu. A threat model-based approach to security testing. *Software: Practice and Experience*, 43(2):241–258, 2013.

[55] Exabeam. Top 8 threat modeling methodologies and techniques. https://www.exabeam.com/information-security/threat-modeling/. Accessed: 2023-04-10.

[56] Qing Li and Yu-Liu Chen. Data flow diagram. In *Modeling and Analysis of Enterprise and Information Systems*, pages 85–97. Springer, 2009.

[57] Nataliya Shevchenko, Timothy A Chick, Paige O'Riordan, Thomas P Scanlon, and Carol Woody. Threat modeling: a summary of available methods. Technical report, Carnegie Mellon University Software Engineering Institute Pittsburgh United . . . , 2018.

[58] Microsoft. Microsoft threat modeling tool - threats. https://docs.microsoft.com/en-us/azure/security/develop/threat-modeling-tool-threats. Accessed: 2023-04-10.

[59] Tony UcedaVelez and Marco M Morana. *Risk Centric Threat Modeling: process for attack simulation and threat analysis*. John Wiley & Sons, 2015.

[60] Paul Saitta, Brenda Larcom, and Michael Eddington. Trike v. 1 methodology document [draft]. *URL: http://dymaxion. org/trike/Trike_v1_Methodology_Documentdraft. pdf*, 2005.

[61] Jackson Wynn. Threat assessment and remediation analysis (tara). Technical report, MITRE CORP BEDFORD MA BEDFORD United States, 2014.

[62] Mina Deng, Kim Wuyts, Riccardo Scandariato, Bart Preneel, and Wouter Joosen. A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. *Requirements Engineering*, 16(1):3–32, 2011.

[63] Bruce Schneier. Attack trees. *Dr. Dobb's journal*, 24(12):21–29, 1999.

[64] Malina Adach, Kaj Hänninen, and Kristina Lundqvist. Security ontologies: A systematic literature review. In *Enterprise Design, Operations, and Computing: 26th International Conference, EDOC 2022, Bozen-Bolzano, Italy, October 3–7, 2022, Proceedings*, pages 36–53. Springer, 2022.

[65] Zareen Syed, Ankur Padia, Tim Finin, Lisa Mathews, and Anupam Joshi. Uco: A unified cybersecurity ontology. 02 2016.

[66] Sean Barnum. Standardizing cyber threat intelligence information with the structured threat information expression (stix™). page 22, 02 2014.

[67] FiRST. Common vulnerability scoring system version 3.1: Specification document. https://www.first.org/cvss/specification-document, 2019. Accessed: 2023-04-10.

[68] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked data: The story so far. In *Semantic services, interoperability and web applications: emerging concepts*, pages 205–227. IGI global, 2011.

[69] Bruno A. Mozzaquatro, Ricardo Jardim-Goncalves, and Carlos Agostinho. Towards a reference ontology for security in the internet of things. In *2015 IEEE International Workshop on Measurements Networking (M N)*, pages 1–6, 2015.

[70] Microsoft. What is vulnerability management? Available at https://www.microsoft.com/en-us/security/business/security-101/what-is-vulnerability-management. Accessed: 2023-04-10.

[71] Bijan Khazai, Tina Kunz-Plapp, Christian Büscher, and Antje Wegner. Vuwiki: An ontology-based semantic wiki for vulnerability assessments. *International Journal of Disaster Risk Science*, 5:55–73, 2014.

[72] Bruno Augusti Mozzaquatro, Carlos Agostinho, Diogo Goncalves, João Martins, and Ricardo Jardim-Goncalves. An ontology-based cybersecurity framework for the internet of things. *Sensors*, 18(9):3053, 2018.

[73] Romilla Syed. Cybersecurity vulnerability management: A conceptual ontology and cyber intelligence alert system. *Information & Management*, 57(6):103334, 2020.

[74] Ju An Wang, Minzhe Guo, Hao Wang, Min Xia, and Linfeng Zhou. Ontology-based security assessment for software products. In *Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies*, pages 1–4, 2009.

[75] Ju An Wang and Minzhe Guo. Ovm: an ontology for vulnerability management. In *Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies*, pages 1–4, 2009.

[76] Jakub Pastuszuk, Patryk Burek, and Bogdan Księżopolski. Cybersecurity ontology for dynamic analysis of it systems. *Procedia Computer Science*, 192:1011–1020, 2021.

[77] Shodan. Search engine for the internet of everything. Available at https://www.shodan.io.

[78] Censys. The censys platform. Available at https://censys.io.

[79] Marc Arnaert, Yoann Bertrand, and Karima Boudaoud. Modeling vulnerable internet of things on shodan and censys: An ontology for cyber security. In *Proceedings of the Tenth International Conference on Emerging Security Information, Systems and Technologies (SECUREWARE 2016)*, pages 299–302, 2016.

[80] MinJu Kim, Sangeeta Dey, and Seok-Won Lee. Ontology-driven security requirements recommendation for apt attack. In *2019 IEEE 27th international requirements engineering conference workshops (REW)*, pages 150–156. IEEE, 2019.

[81] Aaron Steele. Ontological vulnerability assessment. In *Web Information Systems Engineering–WISE 2008 Workshops: WISE 2008 International Workshops, Auckland, New Zealand, September 1-4, 2008. Proceedings 9*, pages 24–35. Springer, 2008.

[82] Rohith Yanambaka Venkata, Patrick Kamongi, and Krishna Kavi. An ontology-driven framework for security and resiliency in cyber physical systems. *ICSEA*, 2018:23, 2018.

[83] Patrick Kamongi, Srujan Kotikela, Krishna Kavi, Mahadevan Gomathisankaran, and Anoop Singhal. Vulcan: Vulnerability assessment framework for cloud computing. In *2013 IEEE 7th international conference on software security and reliability*, pages 218–226. IEEE, 2013.

[84] Patrick Kamongi, Mahadevan Gomathisankaran, and Krishna Kavi. Nemesis: Automated architecture for threat modeling and risk assessment for cloud computing. In *Proc. 6th ASE International Conference on Privacy, Security, Risk and Trust (PASSAT)*, 2014.

[85] Valentina Casola, Alessandra De Benedictis, Carlo Mazzocca, and Rebecca Montanari. Toward automated threat modeling of edge computing systems. In *2021 IEEE International Conference on Cyber Security and Resilience (CSR)*, pages 135–140. IEEE, 2021.

[86] Lars Halvdan Flå, Ravishankar Borgaonkar, Inger Anne Tøndel, and Martin Gilje Jaatun. Tool-assisted threat modeling for smart grid cyber security. In *2021 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)*, pages 1–8. IEEE, 2021.

[87] Michael Smith, Chris Welty, and Deborah Mcguinness. Owl web ontology language guide. 01 2004.

[88] Andrei Brazhuk. Security patterns based approach to automatically select mitigations in ontology-driven threat modelling. 02 2020.

[89] Andrei Brazhuk and Evgeny Olizarovich. Framework for ontology-driven threat modelling of modern computer systems. 02 2020.

[90] Andrei Brazhuk. Towards automation of threat modeling based on a semantic model of attack patterns and weaknesses. *CoRR*, abs/2112.04231, 2021.

[91] Bilal Al Sabbagh and Stewart Kowalski. A socio-technical framework for threat modeling a software supply chain. *IEEE Security & Privacy*, 13(4):30–39, 2015.

[92] P Salini and J Shenbagam. Prediction and classification of web application attacks using vulnerability ontology. *International Journal of Computer Applications*, 116(21), 2015.

[93] Carl E Landwehr, Alan R Bull, John P McDermott, and William S Choi. A taxonomy of computer program security flaws. *ACM Computing Surveys (CSUR)*, 26(3):211–254, 1994.

[94] Robert Luh, Sebastian Schrittwieser, and Stefan Marschalek. Taon: An ontology-based approach to mitigating targeted attacks. In *Proceedings of the 18th International Conference on Information Integration and Web-based Applications and Services*, pages 303–312, 2016.

[95] Eric M Hutchins, Michael J Cloppert, Rohan M Amin, et al. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. *Leading Issues in Information Warfare & Security Research*, 1(1):80, 2011.

[96] MITRE. Att&ck. Available at https://attack.mitre.org.

[97] Ashraf Tantawy, Sherif Abdelwahed, Abdelkarim Erradi, and Khaled Shaban. Model-based risk assessment for cyber physical systems security. *Computers & Security*, 96:101864, 2020.

[98] Victor Bolbot, Gerasimos Theotokatos, Evangelos Boulougouris, and Dracos Vassalos. A novel cyber-risk assessment method for ship systems. *Safety Science*, 131:104908, 2020.

[99] Jean-Marie Flaus. *Cybersecurity of industrial systems*. John Wiley & Sons, 2019.

[100] Farnaz Farzan, Mohsen A Jafari, Dong Wei, and Yan Lu. Cyber-related risk assessment and critical asset identification in power grids. In *ISGT 2014*, pages 1–5. IEEE, 2014.

[101] Jason RC Nurse, Sadie Creese, and David De Roure. Security risk assessment in internet of things systems. *IT professional*, 19(5):20–26, 2017.

[102] Anastasija Collen, Niels A Nijdam, Javier Augusto-Gonzalez, Sokratis K Katsikas, Konstantinos M Giannoutakis, Georgios Spathoulas, Erol Gelenbe, Konstantinos Votis, Dimitrios Tzovaras, Navid Ghavami, et al. Ghost-safe-guarding home iot environments with personalised real-time risk control. 2018.

[103] Information technology — security techniques — information security management — organizational economics. Standard ISO/IEC TR 27016:2014, International Organization for Standardizqation, Geneva, Switzerland, 2014.

[104] Gustavo Gonzalez-Granadillo, Sofia Anna Menesidou, Dimitrios Papamartzivanos, Ramon Romeu, Diana Navarro-Llobet, Caxton Okoh, Sokratis Nifakos, Christos Xenakis, and Emmanouil Panaousis. Automated cyber and privacy risk management toolkit. *Sensors*, 21(16):5493, 2021.

[105] Baidyanath Biswas, Arunabha Mukhopadhyay, Sudip Bhattacharjee, Ajay Kumar, and Dursun Delen. A text-mining based cyber-risk assessment and mitigation framework for critical analysis of online hacker forums. *Decision Support Systems*, 152:113651, 2022.

[106] Halima Ibrahim Kure, Shareeful Islam, and Haralambos Mouratidis. An integrated cyber security risk management framework and risk predication for the critical infrastructure protection. *Neural Computing and Applications*, 34(18):15241–15271, 2022.

[107] Tiago Prince Sales, Fernanda Baião, Giancarlo Guizzardi, João Paulo A Almeida, Nicola Guarino, and John Mylopoulos. The common ontology of value and risk. In *Conceptual Modeling: 37th International Conference, ER 2018, Xi'an, China, October 22–25, 2018, Proceedings 37*, pages 121–135. Springer, 2018.

[108] Ítalo Oliveira, Tiago Prince Sales, Riccardo Baratella, Mattia Fumagalli, and Giancarlo Guizzardi. An ontology of security from a risk treatment perspective. In *Conceptual Modeling: 41st International Conference, ER 2022, Hyderabad, India, October 17–20, 2022, Proceedings*, pages 365–379. Springer, 2022.

[109] Christos Grigoriadis, Adamantios Marios Berzovitis, Ioannis Stellios, and Panayiotis Kotzanikolaou. A cybersecurity ontology to support risk information gathering in cyber-physical systems. In *Computer Security. ESORICS 2021 International Workshops: CyberICPS, SECPRE, ADIoT, SPOSE, CPS4CIP, and CDT&SECOMANE, Darmstadt, Germany, October 4–8, 2021, Revised Selected Papers*, pages 23–39. Springer, 2022.

[110] Carmen Sánchez-Zas, Víctor A Villagrá, Mario Vega-Barbas, Xavier Larriva-Novo, José Ignacio Moreno, and Julio Berrocal. Ontology-based approach to real-time risk management and cyber-situational awareness. *Future Generation Computer Systems*, 141:462–472, 2023.

[111] Oluwasefunmi T Arogundade, Adebayo Abayomi-Alli, and Sanjay Misra. An ontology-based security risk management model for information systems. *Arabian Journal for Science and Engineering*, 45:6183–6198, 2020.

[112] Mahmoud Khaleghi, Mohammad Reza Aref, and Mehdi Rasti. Context-aware ontology-based security measurement model. *Journal of Information Security and Applications*, 67:103199, 2022.

[113] Fabio De Rosa, Nicolò Maunero, Luca Nicoletti, Paolo Prinetto, and Martina Trussoni. Ontology for cybersecurity governance of ict systems. In *ITASEC22–Italian Conference on Cybersecurity, June 20th-23rd*, pages 1–12, 2022.

[114] W3C OWL Working Group. Owl 2 web ontology language document overview (second edition). https://www.w3.org/TR/owl2-overview/. Accessed: 2023-04-10.

[115] RDF Working Group. Resource description framework (rdf). Available at https://www.w3.org/RDF/, 2014. Accessed: 2023-04-10.

[116] Available at https://protege.stanford.edu/.

[117] Harold Boley, Mike Dean, Benjamin Grosof, Ian Horrocks, Peter F. Patel-Shcneider, and Said Tabet. Swrl: A semantic web rule language combining owl and ruleml. https://www.w3.org/Submission/SWRL/. Accessed: 2023-04-10.

[118] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical owl-dl reasoner. *Journal of Web Semantics*, 5(2):51–53, 2007.

[119] N. Noy and Deborah Mcguinness. Ontology development 101: A guide to creating your first ontology. *Knowledge Systems Laboratory*, 32, 01 2001.

[120] Matthew Horridge, Simon Jupp, Georgina Moulton, Alan Rector, Robert Stevens, and Chris Wroe. A practical guide to building owl ontologies using protégé 4 and co-ode tools edition1. 2. *The university of Manchester*, 107, 2009.

[121] Thomas R Gruber. Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, 43(5-6):907–928, 1995.

[122] Pari Delir Haghighi, Frada Burstein, Arkady Zaslavsky, and Paul Arbon. Development and evaluation of ontology for intelligent decision support in medical emergency management for mass gatherings. *Decision Support Systems*, 54(2):1192–1204, 2013.

[123] Asunción Gómez-Pérez. Evaluation of ontologies. *International Journal of intelligent systems*, 16(3):391–409, 2001.

[124] Jonathan Yu, James A Thom, and Audrey Tam. Evaluating ontology criteria for requirements in a geographic travel domain. In *On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2005, Agia Napa, Cyprus, October 31-November 4, 2005, Proceedings Part II*, pages 1517–1534. Springer, 2005.

[125] Henk Jonkers, Erik Proper, and Mike Turner. Togaf™ and archimate®: A future together. *White Paper W*, 192, 2009.

[126] Archimate® 3.2 specification. Standard C226, National Institute of Standard and Technology, Gaithersburg, USA, 2022.

[127] Nan Messe, Vanea Chiprianov, Nicolas Belloir, Jamal El-Hachem, Régis Fleurquin, and Salah Sadou. Asset-oriented threat modeling. In *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 491–501. IEEE, 2020.

[128] MITRE. Cve glossary. Available at https://www.cve.org/ResourcesSupport/Glossary#.

[129] Brett Crawley Jonathan. Capec-stride mapping. https://ostering.com/blog/2022/03/07/capec-stride-mapping/. Accessed: 2023-04-10.

[130] Hanen Abbes and Faiez Gargouri. M2onto: an approach and a tool to learn owl ontology from mongodb database. In *International Conference on Intelligent Systems Design and Applications*, pages 612–621. Springer, 2016.

[131] MITRE. Mapping mitre att&ck® to cves for impact. https://github.com/center-for-threat-informed-defense/attack_to_cve, 2022.

[132] Edmond Jajaga and Lule Ahmedi. C-swrl: Swrl for reasoning over stream data. In *2017 IEEE 11th International Conference on Semantic Computing (ICSC)*, pages 395–400, 2017.

[133] Yongzhong He, Yiming Wang, Sencun Zhu, Wei Wang, Yunjia Zhang, Qiang Li, and Aimin Yu. Automatically identifying cve affected versions with patches and developer logs. *IEEE Transactions on Dependable and Secure Computing*, pages 1–15, 2023.

[134] Yuning Jiang, Manfred Jeusfeld, and Jianguo Ding. Evaluating the data inconsistency of open-source vulnerability repositories. In *Proceedings of the 16th International Conference on Availability, Reliability and Security*, pages 1–10, 2021.

[135] Kenta Kanakogi, Hironori Washizaki, Yoshiaki Fukazawa, Shinpei Ogata, Takao Okubo, Takehisa Kato, Hideyuki Kanuka, Atsuo Hazeyama, and Nobukazu Yoshioka. Tracing cve vulnerability information to capec attack patterns using natural language processing techniques. *Information*, 12(8):298, 2021.

[136] Giulio Berra, Gaspare Ferraro, Matteo Fornero, Nicolo Maunero, Paolo Prinetto, and Gianluca Roascio. Paideusis: A remote hybrid cyber range for hardware, network, iot security training. In *ITASEC*, pages 284–297, 2021.

[137] Wei Hu, Chip-Hong Chang, Anirban Sengupta, Swarup Bhunia, Ryan Kastner, and Hai Li. An overview of hardware security and trust: Threats, countermeasures, and design tools. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 40(6):1010–1038, 2020.

[138] Wenjun Xiong, Emeline Legrand, Oscar Åberg, and Robert Lagerström. Cyber security threat modeling based on the mitre enterprise att&ck matrix. *Software and Systems Modeling*, 21(1):157–177, 2022.

[139] Tom Butler, Daniel Gozman, and Kalle Lyytinen. The regulation of and through information technology: Towards a conceptual ontology for is research, 2023.