Politecnico di Torino

Master of Science in Cybersecurity

Master Degree Thesis

An Attack Risk Assessment Model for Network Security Automation

Supervisors
prof. Daniele Bringhenti
prof. Fulvio Valenza
prof. Riccardo Sisto

Candidate
Francesca Coriale

Academic Year 2024-2025

Summary

Next-generation computer networks are increasingly complex due to their size, heterogeneity, and dynamism, making them vulnerable to sophisticated, multi-stage and multi-vector attacks. Manual security reconfiguration methods are too slow and error-prone, leading to delays and misconfigurations. To address these problems, the recent literature on network security configuration has focused on automated reconfiguration approaches for faster and more resilient responses. However, the dynamic nature of modern computer networks impacts the preservation of security during the updates, as improper sequencing can create insecure transient states. In lights of these motivations, the FATO methodology has been proposed. The aim is to optimize the scheduling of configuration changes in distributed virtual firewalls to maximize the number of secure intermediate states, minimizing violations of prioritized security policies. Optimality and formal correctness are achieved throughout the formulation of a Maximum Satisfiability Modulo Theories (MaxSMT) problem.

Although FATO is one of the best approaches for automatic firewall reconfiguration, it presents some critical limitations. On one hand, its reconfiguration mechanism is not customized by attack types, since it adopts a general-purpose optimization strategy, which is not always adequate. On the other hand, it lacks a detailed analysis of its applicability to specific contexts.

This thesis aims to address the aforementioned limitations of FATO by investigating how it can be improved to mitigate real complex attacks and provide higher resilience. In order to achieve this objective, the contribution of this work consists in the proposal and definition of an extension for FATO tailored to mitigate complex and multi-vector attacks in a more efficient and responsive way.

To support this objective, an in-depth analysis of the Risk Assessment Model is conducted, and a set of enhancements is proposed to make it reactive to ongoing attacks. The revised model introduces a dynamic prioritization mechanism that evaluates each attack based on two key factors: the immediate impact it has on the system and the likelihood of its recurrence. By combining these two dimensions, the model computes a risk score that reflects both the severity and the persistence of each threat. This score is then used to inform the reconfiguration strategy, ensuring that the most harmful and frequent threats are addressed with higher urgency. The result is a more context-aware firewall reconfiguration process, capable of adapting its priorities in real-time to effectively mitigate the evolving threat landscape.

The effectiveness of the enhanced reconfiguration model is validated through its application to real-life scenarios involving complex and multi-vector attacks. Specifically, the model was tested in simulated environments that accurately replicate

the conditions of modern enterprise networks under coordinated attack campaigns. These scenarios included combinations of lateral movement, privilege escalation, and DDoS tactics, reflecting the sophistication of current threat actors. The evaluation demonstrated that the proposed extension to FATO significantly improved the system's ability to respond more quickly and accurately to the evolving attack patterns. Compared to the original approach, the enhanced model consistently achieved better results in terms of reducing policy violations, minimizing exposure windows, and prioritizing reconfiguration actions based on risk.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Thesis introduction

In today's landscape, large-scale systems are growing exponentially, and their increasing complexity is driven by several factors, including the rapid growth in the number of connected devices, the diversification of network types, and the evolving demands of users, all of which contribute to a more dynamic and challenging security environment. To cope with these challenges, new processes have been adopted, such as *Software-Defined Networking* and the "softwarization" of networks' components, which have allowed the introduction of new paradigms, but at the same time, new points of exposure. As a consequence, attackers grow more skilled and aware of the potential vulnerabilities to exploit, as well as the immense consequences their actions produce, making the need for adaptive and resilient security solutions more urgent than ever. The heterogeneity of network devices complicates the identification of all potential vulnerabilities, and the wide range of attack types further challenges effective network management. As a result, network security plays a crucial role in neutralizing these vulnerabilities with adequate defence.

Many organizations still rely on the manual configuration of their security functions. However, manually configuring complex systems like firewalls, IDS, and NAT becomes extremely hard and time-consuming, even in small networks, increasing the likelihood of misconfigurations [1]. To address this issue, automation has been recently leveraged by research to improve the state of the art of network security configuration, as the works [2],[3] exhaustively analysed. The main goal is to minimize human intervention, guaranteeing scalability and flexibility. Additionally, an automated process can also be combined with optimization techniques and formal verification to avoid unnecessary resource consumption and to identify or prevent configuration mistakes.

Deeply linked to this philosophy, the *VEREFOO (VErified REFinement and Optimized Orchestration)* framework has the purpose to automatically allocate and configure Network Security Functions (NSFs), specifically packet filters, which are the most common firewall technology used in computer networks [4]. VEREFOO is able to combine full automation with optimality and formal verification, even for large networks. Given its relevance, several variations have been implemented to respond to different problems. The evolution described in [5] responds to the matter

of the enormous energy consumption resulting from the reconfiguration of NSFs on large-scale networks. While the version proposed in [6] extends the framework to other Network Security Functions, putting the focus on the configuration of VPN systems.

Even though VEREFOO remains one of the best frameworks for the automatic configuration of network systems, it results computationally inefficient for reconfiguring an already deployed network. In order to overcome this limitation, another framework has been proposed, called *React-VEREFOO*. React-VEREFOO represents a novel approach for the reconfiguration of a distributed firewall system, which combines full automation, optimization of resource consumption, and formal correctness assurance of the computed configuration. This tool is designed to generate a reconfigured version of the current settings in the shortest time possible, while ensuring its formal correctness and scalability, as exhaustively validated in [7].

While React-VEREFOO efficiently sorts out the computational burden that a reconfiguration from scratch involves, it still reports limitations when a distributed security function is subject to a series of configuration changes. In this scenario, the transient from the application of the first change to the last one may present insecure temporary states, where the required security protection is missing. Each state consists of the deployment or the removal of a virtual function or updating the function rules. The time required for these operations may not be negligible, and the resulting security gap can lead to disruptions of essential services, as well as undetected intrusions and exploitations of temporary vulnerabilities by external attackers. To respond to this problem, the paper [8] proposes a new framework for the automatic scheduling of the intermediate stages, called *FATO (FirewAll Transients Optimizer)*. The aim is to minimise the number of insecure states, thus limiting the period of time during which the system may be at risk. Similar to React-VEREFOO, it works with packet filters and establishes their reconfiguration order. However, FATO is based on priority weights associated with each connectivity policy, which represent security policies for communications in the network.

The evolution of modern attacks, increasingly more sophisticated with combinations of vectors or techniques, has highlighted the need to define a priority between the various reconfiguration phases, not only based on the relationships between connectivity policies but also on the impact of each attack event. This would allow to quickly restore system security with the best strategy possible and in the shortest time. For this reason, it is necessary to take a look at the risk assessment branch, which makes risk calculation its founding element. Well-known models, such as the one proposed and validated by NIST [9], allow to obtain well-defined and verifiable priorities for different attacks. This consents organizations to formulate their defence efficiently and without leaving crucial points uncovered. By combining factors from different categories, such as economic, reputational, and operational impact, attack frequency, and the effectiveness of control and defence systems already on site, these models are able to establish complete and efficient risk levels.

After all these considerations, this thesis wants to expand the application of FATO to large-scale systems, in critical contexts, by readjusting known risk assessment models. The goal is to provide a semi-quantitative analysis of the risk

associated to each vector or stage of the attack in order to define the corresponding priority values and to maximize the number of secure states. To do so, a new model for the reconfiguration of packet filter firewalls in large-scale networks has been studied. The proposed model has been validated as a part of the FATO framework in critical infrastructures scenario.

1.2 Thesis description

The remaining of the thesis is structured in the following way:

- **Chapter 2** provides a background on risk assessment, focusing on the models developed by NIST and Open FAIR. These models are presented and explained in detail to offer a structured understanding of how risks can be identified, analysed, and quantified within network security contexts.
- **Chapter 3** provides a background on network security automation, discussing its benefits and challenges, and explores the current state of the art. It also presents an overview of the React-VEREFOO framework and the FATO framework, setting the foundation for the thesis.
- **Chapter 4** explains the objective of this thesis, introducing the general idea about the approach and the main elements of the work done for reaching the goal.
- **Chapter 5** presents the main contributions of this thesis. The core of the approach is a model for computing the risk associated with a singular attack event, taking into account both its impact and likelihood of recurrence. The chapter details the metrics used for this computation and explains how the proposed model was derived, building upon the foundational concepts introduced in the NIST and Open FAIR models.
- **Chapter 6** provides an explanation of the implementation of the model in the FATO framework and the tested critical infrastructure scenario, presenting the achieved results with a reflection about what objectives have been achieved.
- **Chapter 7** contains the conclusion, which summarize the achieved goals of this thesis and some possible path that could be followed in future work to improve the solution.

Chapter 2

Risk Assessment

This chapter provides a comprehensive overview of some of the key technical concepts essential for understanding the solutions and findings discussed in the remainder of this document. A particular focus is placed on cybersecurity risk assessment, its fundamental components, their interaction, and the role they play in producing meaningful outcomes. Additionally, the chapter explores and compares the various models and methodologies currently adopted across the industry. Finally, the traditionally most used frameworks to compute the risk score will also be analysed, and their effectiveness and limitations will be discussed.

In the context of cybersecurity, the term *risk* refers to the "risk of financial loss, operational disruption, or damage, from the failure of the digital technologies employed for informational and/or operational functions introduced to a manufacturing system via electronic means from the unauthorized access, use, disclosure, disruption, modification, or destruction of the manufacturing system" [10]. Risk assessment is one part of the broader security task of risk management. As reported by NIST, risk management is "the process of managing risks to organizational operations (including mission, functions, image, or reputation), organizational assets, or individuals resulting from the operation of an information system, and includes: (i) the conduct of a risk assessment; (ii) the implementation of a risk mitigation strategy; and (iii) employment of techniques and procedures for the continuous monitoring of the security state of the information system" [11], as also shown in Figure 2.1.

In this frame, the risk assessment is the phase where the risks to the organizational operations, assets or people are analysed and evaluated. It includes threat and vulnerability analyses, and considers the effectiveness of security controls already in place. The primary purpose of risk assessment is to support organizations in allocating an appropriate budget for cybersecurity measures. Within the constraints of this budget, administrators can then prioritize and implement security controls that offer the most effective protection. This process involves estimating both the potential impact of security incidents and the likelihood of their occurrence, thereby enabling a more informed and strategic approach to manage cybersecurity risks.

As fully explained by William Stallings in his book *Effective Cybersecurity: A Guide to Using Best Practices and Standards* [12], the risk assessment process can

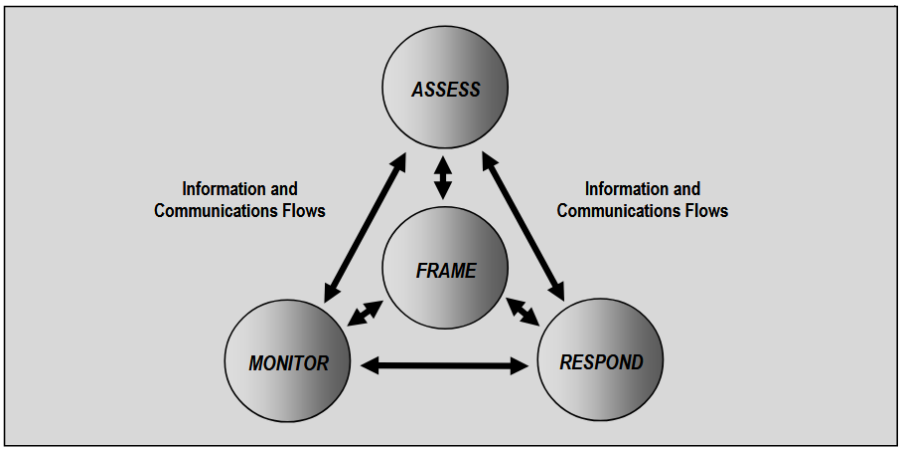

Figure 2.1. NIST Risk Management Frame

generally be defined by the diagram shown in Figure 2.2.

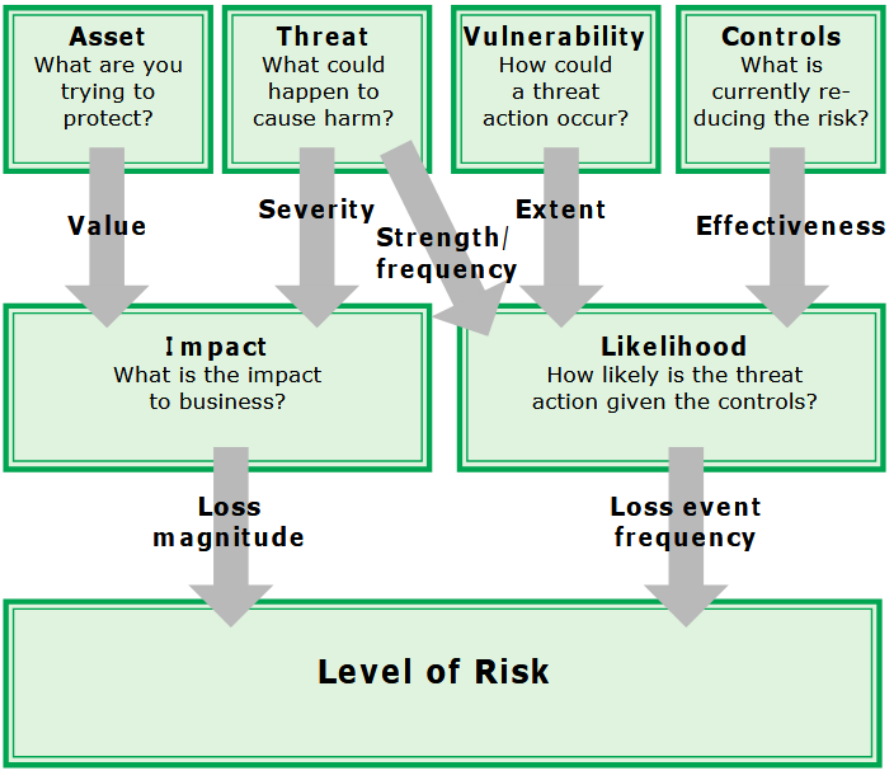

Figure 2.2. Risk Assessment Diagram

The final result represented by the *Level of Risk* is the magnitude of a risk, or a combination of risks, expressed in terms of repercussions and their likelihood. It serves as a metric that an organization can use to assess the need and the expected cost of taking remedial measures through risk treatment.

Two main assessments should be pursued in parallel: *Loss Magnitude* and *Loss Event Frequency*. The first one (also referred to as *Impact*) represents the estimated severity of consequences that may result from a successful exploitation of

a vulnerability by a threat. Loss Magnitude can be decomposed into different aspects, depending on the organizational context and the risk assessment framework being used. Typical categories include:

- Financial Loss, including costs incurred due to theft, fraud, regulatory fines, or loss of revenue.
- Operational Impact resulting from disruptions to business processes, production downtime, or degraded performance.
- Reputational Damage, with consequent loss of stakeholder and customer trust, and negative media coverage.
- Legal and Regulatory Consequences, due to non-compliance with standards or laws.
- Safety Risks, especially in safety-critical environments, where cyber incidents may lead to physical harm.

Understanding the potential loss magnitude allows organizations to make informed decisions about the cost-effectiveness of risk treatments and resource allocation.

The second one refers to the estimated number of times a threat event is expected to result in a loss over a given time period. Understanding the Loss Event Frequency (also referred to as *Likelihood*) enables organizations to assess how often they might expect damaging incidents, which in turn supports prioritization of security controls and risk mitigation strategies.

Going into more detail, the Impact factor is generally derived from the combination of the following two elements:

- Asset, which includes elements directly tied to information processing, such as data, devices, and other components that enable or support information-related functions. These assets may be subject to unauthorized access, use, disclosure, alteration, destruction, or theft, potentially resulting in loss. Additionally, the asset category includes elements such as organizational knowledge, reputation, and public image.
- Threat, which refers to any circumstance or event with the potential to adversely impact organizational operations (including mission, functions, image, or reputation), organizational assets, or individuals through an information system via unauthorized access, destruction, disclosure, modification of information, and/or denial of service. A threat can be analysed from different points of view, but to determine the impact it is sufficient to focus on its *severity*, i.e. the magnitude of the damage that the threat can cause.

On the other side, the Likelihood factor can be obtained considering three main elements:

- Threat, specifically, its *strength* and *frequency* understood as the probable level of force that a threat agent is capable of applying against an asset and the probable frequency, within a given time frame, that a threat agent will act against an asset.

- Vulnerability, which can be seen as any weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source. The *extent* of a vulnerability is a measure of how significant the weakness is.

- Controls, which are safeguards or countermeasures implemented within an information system or organization to protect the confidentiality, integrity, and availability or other security properties of an asset. They are designed to address specific security requirements and reduce risk to an acceptable level. Their *effectiveness* is a measure of how successful the controls are in blocking adversaries.

Then the model determines the likelihood that a threat will cause harm given the threat action, the vulnerability exploited and the effectiveness of the security controls that are in place. While similar to threat event frequency, loss event frequency differs in that it only accounts for events in which the threat successfully causes harm or loss.

An explicative example shown by Stallings is the following one: a hacker (threat agent) may exploit known vulnerabilities (vulnerability) in a remote authentication protocol (vulnerability target) to disrupt (policy violated) remote authentication (asset exposed). The threat is unauthorized access. The assets are anything that can be compromised by an unauthorized access. The vulnerability expresses how a threat action could occur (for example, by access through a web interface). Existing security controls for this vulnerability reduce the likelihood of a threat action.

Organizations can treat the two factors of risk assessment, impact and likelihood, either *qualitatively* or *quantitatively*. In the first case the approach is not structured and it highly depends on the expertise of the administrator in charge. This method allows for rapid evaluation without the need for extensive data collection or statistical modelling. Even though these types of approaches are very versatile, they are naturally predisposed to the dangers of incorrect interpretation. The impact and likelihood scores are not numerical but chosen by security administrators based on their subjectivity. In this context, organizations can rely on the three security categories (low, moderate, high) defined by FIPS 199 (Standards for Security Categorization of Federal Information and Information Systems [13]). The overall risk level is then determined using a matrix that combines impact and likelihood, as illustrated in Figure 2.3.

The other approach, the quantitative one, relies on a more scientific and methodical process. Quantitative methods generally apply mathematical or statistical models, using data from real-world event measurements to provide arithmetical estimates of risk. By assigning numerical values to risk factors, the assessment results more objective, allowing for quicker and clearer prioritization. However, this method can be challenging to implement due to the frequent lack of sufficient data

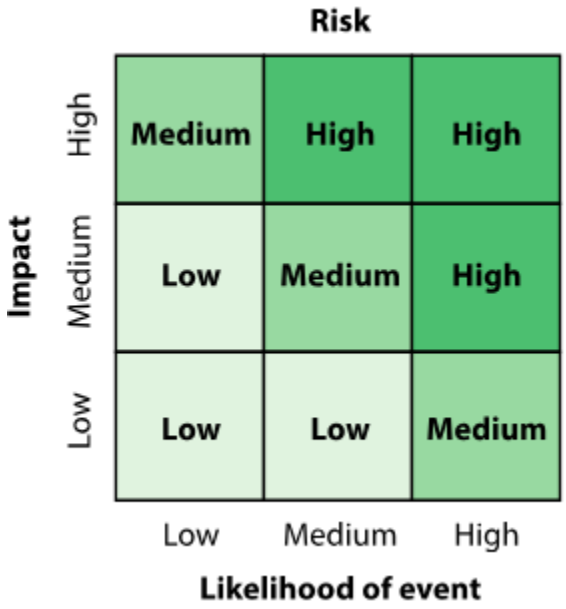

Figure 2.3. Qualitative Risk Assessment matrix

for accurate estimation and the difficulty of adapting the model to different use cases. In this approach, the final risk value can be efficiently computed as the combination of the cost and the likelihood of the occurring threat, with the following equation (2.1):

$$\text{Risk} = \text{Loss Magnitude} \times \text{Loss Event Frequency} \tag{2.1}$$

A final, hybrid approach is the *semi-quantitative* assessment, provides the benefits of both methods proposed above. In this approach, expert evaluations are translated into numerical values using bins, scales or representative numbers, enabling more precise comparisons across various risk scenarios. When scales or categories offer sufficient granularity, relative prioritization among results is better supported compared to a purely qualitative approach. Moreover, scales such as 1 to 10 can be easily translated to qualitative terms, making them useful for communicating risk levels to non-technical personnel.

2.1 NIST Risk Assessment

A well-known example of *qualitative* risk assessment is the process proposed by NIST in its "Guide for Conducting Risk Assessments" [9], through which organizations have a step-by-step guide on:

1. preparing for risk assessments
2. conducting risk assessments
3. communicating risk assessments

4. maintaining risk assessments over time.

A crucial part of risk assessment methodologies is the risk assessment process, where the risk model and the assessment approach are identified. The identification of the risk model allows the definition of the risk factors to be assessed and the relationships among them. In the NIST Risk Assessment framework, the chosen risk model is the one in Figure 2.4. Unlike the diagram presented by Stallings, here the key concepts are a bit different.

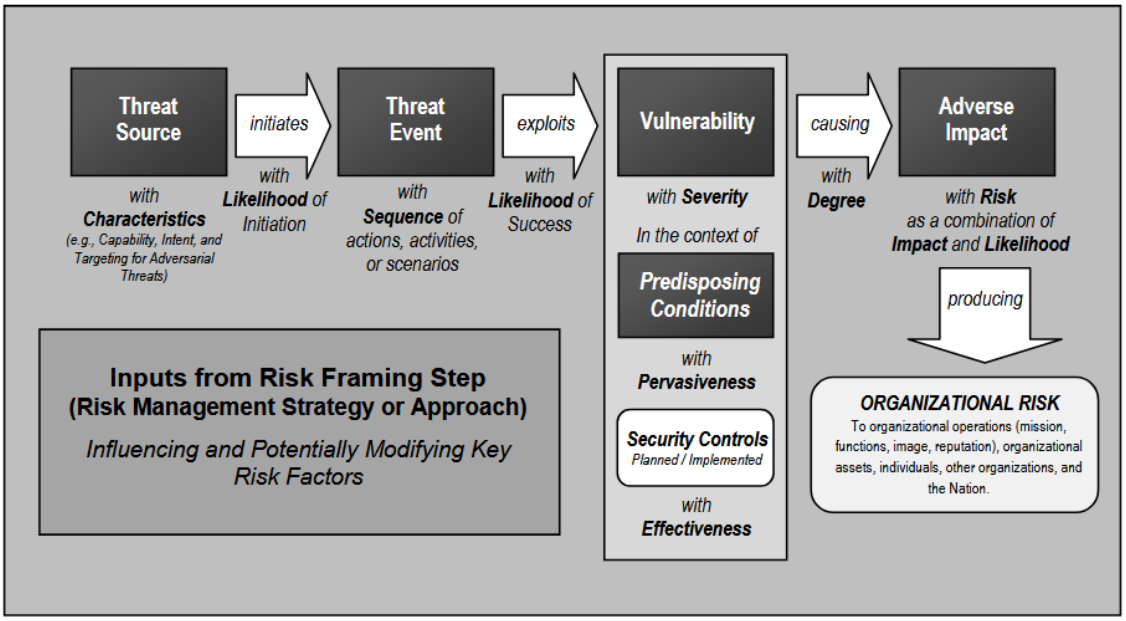

Figure 2.4. NIST Risk Model

The *threat* factor is decomposed into two main elements: event and source. A threat source is defined as "the intent and method targeted at the exploitation of a vulnerability or a situation and method that may accidentally exploit a vulnerability" [9]. Types of threat sources include hostile cyber or physical attacks, human errors and structural failures of organization's resources. It is important to note that multiple threat sources can initiate the same threat event, for example, a data breach could be triggered by a malicious insider, an external hacker, or even an unintentional misconfiguration by an employee. Having a deeper knowledge about the threat sources gives organizations a better understanding of the capabilities and the intentions behind the attacks. This helps to narrow the set of threat events that are most relevant. When threat events are identified with a high level of specificity, they can be modelled into threat scenarios, defined as a set of related events attributed to one or more threat sources that ultimately lead to adverse outcomes.

The *vulnerability* factor does not only identify vulnerability within information systems, but also those found in organizational governance structures or in external relationships, such as supply chains or telecommunications providers. In general, risks emerge from a sequence of threat events, each of them exploiting one or more existing vulnerabilities. This is why the development of threat scenarios

may clarify how a set of vulnerabilities could be exploited by one or more threat events, since some vulnerabilities may not be exposed unless other vulnerabilities are exploited. The vulnerability severity can be determined by the "extent of the potential adverse impact if such a vulnerability is exploited by a threat source" [9], thus being context-dependent. Vulnerabilities often emerge or are influenced within the context of outdated technologies, weaknesses in information systems and failover mechanisms, all falling into the category of *predisposing conditions*. A predisposing condition refers to a factor present within an organization, its business processes, enterprise architecture, information systems, or operating environment that influences the likelihood that a threat event, once triggered, will lead to adverse effects on organizational operations, assets or individuals. For instance, an isolated information system with no external network access has a reduced likelihood of being targeted by network-based cyber attacks.

The *likelihood* of the occurrence factor combines an estimate of the likelihood that the threat event will be initiated with an estimate of the likelihood of the impact, meaning the likelihood that the threat event results in adverse impacts. Similarly to the concept described by Stallings, it is the result of an "analysis of the probability that a given threat is capable of exploiting a given vulnerability, or a set of vulnerabilities, with respect to a given time frame" [9]. In addition, it is based on the adversary intent, capabilities and targeting, taking into consideration predisposing conditions and the effectiveness of already deployed security controls to detect and limit damage. The likelihood factor addresses the probability of adverse impact, regardless of the magnitude of harm that can be expected.

This last concept is represented by the *impact* factor. Organizations explicitly define how their priorities and values influence the identification of high-value assets and the assessment of potential adverse impacts on organizational stakeholders. In cases where such definitions are not documented, these priorities and values can often be inferred from strategic plans and internal policies. For example, security categorization levels help determine the potential impact of compromising different types of information, while Privacy Impact Assessments highlight the consequences of data destruction, corruption, or loss of accountability for information resources.

Finally, the organizational *risk* result matches the one defined by Stallings, as function of the likelihood of a threat event's occurrence and its potential adverse impact.

To effectively assess each of the factors within the NIST Risk Assessment framework, organizations must rely on a variety of information sources, either internal or external to the organization. When relying on external information sources, organizations must consider potential limitations such as lack of specificity, delayed reporting, or reduced relevance to their particular threat landscape or operational environment. External threat intelligence may offer broad insights into adversary tactics or emerging vulnerabilities but often lacks the contextual detail necessary for accurate prioritization. Therefore, such data should be supplemented with internal monitoring, asset-specific analysis, and organizational knowledge to ensure risk assessments remain both timely and tailored to the single entity exposure. For the threat component, relevant data can be obtained from internal security logs, monitoring results, past incident reports, industry-specific threat advisories, but also

from public information sharing platforms such as ISACs (Information Sharing and Analysis Centers), research and nongovernmental organizations.

NIST classifies threat sources into four main categories: adversarial, accidental, structural, and environmental. Adversarial sources include threat actors with intent and capability to cause harm, such as nation-state actors, cybercriminals, hacktivists, and malicious insiders. Accidental sources typically involve unintentional human errors, such as misconfigurations or mishandling of sensitive data, while structural sources relate to failures of hardware, software, or supporting infrastructure. Representative examples of adversarial threat events include information gathering, delivering of malicious capabilities (e.g. deliver targeted malware for control of internal systems and exfiltration of data), denial-of-service and distributed denial-of-service attacks targeting availability, exploitation of vulnerabilities on internal organizational information systems, or supply chain attacks exploiting third-party software vulnerabilities. Vulnerability information is typically drawn from vulnerability assessment reports, penetration testing reports, configuration audits, or public databases such as the National Vulnerability Database (NVD).

Predisposing conditions may be identified through organizational architecture documentation, environmental assessments, enterprise architecture, and common infrastructure. NIST classifies the predisposing conditions into three categories: information-related, technical, and operational-environmental. The first one refers to those conditions where there is the need to handle information (as it is created, transmitted, stored, processed, and/or displayed) in a specific manner due to its sensitivity (or lack of sensitivity), legal or regulatory requirements, or other organizational agreements. Technical conditions involve both architectural aspects, such as use of specific products or product lines or the allocation of specific security functionality to common controls, and functional aspects, for example limited system functionality (e.g. constrained communications, sensor capabilities, or embedded controller operations). Operational-environmental conditions refer to the extent to which an organization can depend on physical, procedural, and personnel-based safeguards within its operating environment to mitigate risks. These may include access controls, facility security measures, well-defined operational procedures, or the presence of trained personnel. Organizations assess the overall likelihood of threat events by using inputs from tables reporting the likelihood of initiation for adversarial (or non-adversarial) threat events and the likelihood of threat events having adverse impacts if initiated. Any specific algorithm or rule for combining the determined likelihood values depends on: the general organizational attitudes toward risk, including overall risk tolerance and tolerance for uncertainty, the specific tolerances toward uncertainty in different risk factors, and the organizational weighting of risk factors. Depending on these considerations, organizations might adopt one of the following possible approaches:

- use the maximum/minimum of the two likelihood values
- consider only the likelihood of initiation/occurrence, assuming that if threat events are initiated/occur, they will result in adverse impacts
- consider only the likelihood of impact, assuming that if threat events could result in adverse impacts, then for sure adversaries will initiate them

- take a weighted average of the two likelihood values.

An example of qualitative matrix for the likelihood factor is exposed in Figure 2.5.

TABLE G-5: ASSESSMENT SCALE – OVERALL LIKELIHOOD

Likelihood of Threat Event Initiation or Occurrence	Likelihood Threat Events Result in Adverse Impacts				
	Very Low	**Low**	**Moderate**	**High**	**Very High**
Very High	Low	Moderate	High	Very High	Very High
High	Low	Moderate	Moderate	High	Very High
Moderate	Low	Low	Moderate	Moderate	High
Low	Very Low	Low	Low	Moderate	Moderate
Very Low	Very Low	Very Low	Low	Low	Low

Figure 2.5. NIST likelihood qualitative matrix

The impact factor may derive from business continuity plans, asset criticality ratings, Privacy Impact Assessments, legal and regulatory requirements, and security categorization processes. Ultimately, the overall risk is synthesized by combining these inputs to support decision-making, prioritization, and continuous improvement of the cybersecurity posture. The final qualitative matrix is exposed in Figure 2.6. Organizations will order the threat events register following in descending order the risk level values associated with each of them.

TABLE I-2: ASSESSMENT SCALE – LEVEL OF RISK (COMBINATION OF LIKELIHOOD AND IMPACT)

Likelihood (Threat Event Occurs and Results in Adverse Impact)	Level of Impact				
	Very Low	**Low**	**Moderate**	**High**	**Very High**
Very High	Very Low	Low	Moderate	High	Very High
High	Very Low	Low	Moderate	High	Very High
Moderate	Very Low	Low	Moderate	Moderate	High
Low	Very Low	Low	Low	Low	Moderate
Very Low	Very Low	Very Low	Very Low	Low	Low

Figure 2.6. NIST risk qualitative matrix

2.2 Open FAIR Risk Analysis

A well-known example of *quantitative* risk assessment is the standard proposed by the Open Group organization in its "Open FAIR Body of Knowledge". The Open Group is a global consortium that develops open, vendor-neutral technology standards and certifications. The FAIR (Factor Analysis of Information Risk) framework is one of its most relevant contributions to risk management. FAIR provides a quantitative model for understanding, analysing, and measuring information risk in financial terms. It helps organizations make well-informed decisions about risk prioritization and mitigation based on estimated values of loss magnitude and event frequency. The Open FAIR Body of Knowledge is mainly composed by two contributions: the Open Group Risk Analysis (O-RA) Standard [14] and the Open Group Risk Taxonomy (O-RT) Standard [15]. The first one provides standards for different aspects of information security risk analysis, the other defines a taxonomy for the factors that drive information security risk.

A first fundamental clarification concerns the scope covered by the Open FAIR framework with respect to the NIST's: a risk analysis framework, as the one developed by the Open Group, "use measurements and estimates of risk factors to provide an overall statement on the probable frequency and probable magnitude of future loss". In contrast, a risk assessment framework, such as the NIST's framework, covers a broader context including the processes and technologies used to identify, evaluate, and communicate risks. In other words, Open FAIR's risk analysis approach does not contradict the whole NIST risk assessment framework, but rather offers an alternative to the qualitative analysis component within it, by introducing a more structured and quantitative methodology.

The Open FAIR risk analysis model supports a top-down approach, enabling analysts to derive high-level risk factors indirectly when direct measurement is not feasible. In such cases, estimates can be constructed from more accessible, lower-level sub-factors, ensuring flexibility and adaptability in the analysis process. The diagram shown in Figure 2.7 depicts the model explained in detail in the O-RA documentation.

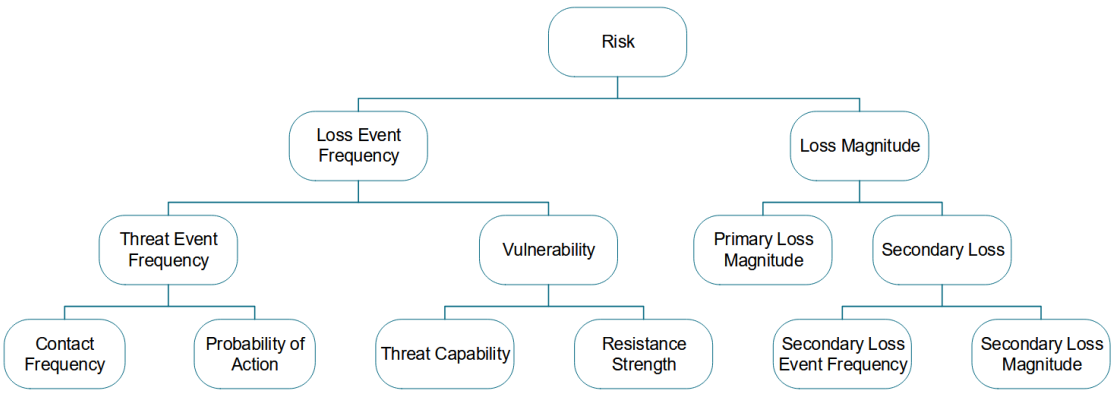

Figure 2.7. Open FAIR risk analysis diagram

The left side of Figure 2.7 shows the factors that allow the evaluation of Loss

Event Frequency (LEF). The LEF estimate reflects how many times the Loss Scenario is expected to occur in a given timeframe. Loss Scenario refers to "the story of loss that forms a sentence from the perspective of the Primary Stakeholder". At the higher level, LEF can be derived from the combination of Threat Event Frequency and Vulnerability. The Threat Event Frequency is "the probable frequency, within a given timeframe, that a Threat Agent will act against an Asset" breaching or impairing the asset's confidentiality, integrity or availability, while Vulnerability is meant to be "the probability that a Threat Event becomes a Loss Event". The estimation of Threat Event Frequency depends on two primary factors: the Contact Frequency, which measures how often a Threat Agent interacts with an Asset, and the Probability of Action, which represents the probability that the Threat Agent will act against the Asset. For a Loss Event to take place, initial contact between the Threat Agent and the Asset must occur. Following this contact, the occurrence of a Threat Event is determined by whether the Threat Agent chooses to act. If no action is taken despite the contact, a Threat Event does not occur, thus not every instance of contact leads to a resulting threat. The probability that a Threat Agent acts against the Asset is affected by the attacker's perception of the benefits and costs the action would entail, in particular by the perceived value of the Asset, based on their motivations, the perceived level of effort required to accomplish the task and the perceived risk of negative consequences. Contact between the Agent and the Asset can be physical or "logical", meaning via networks, and in both cases can be occur in three different modes:

- Random, when the Threat Agent finds the Asset incidentally while engaged in general or undirected activities
- Regular, if the contact occurs because of the regular actions of the Threat Agent
- Intentional, the Threat Agent deliberately searches for specific targets.

If it is possible to obtain data on Threat Events and Loss Events, in a given framework, then the Vulnerability factor is simply the conditional probability that the Threat Event will become a Loss Event expressed in the equation 2.2.

$$\text{Vulnerability} = Pr(\text{Loss Magnitude} \mid \text{Loss Event Frequency}) \tag{2.2}$$

Otherwise, the Vulnerability factor can be estimated as combination of other two sub-factors: the Threat Capability, meaning the "probable level of force (as embodied by the time, resources, and technological capability) that a Threat Agent is capable of applying against an Asset", and the Resistance Strength, referring to the "strength of a Control as compared to the probable level of force that a Threat Agent is capable of applying against an Asset". In this case, the Vulnerability factor is the conditional probability that the Threat Capability is greater than the Resistance Strength, as shown in the formula 2.3.

$$\text{Vulnerability} = Pr(\text{Threat Capability} > \text{Resistance Strength}) \tag{2.3}$$

Threat Capability represents the range of a threat agent's skills, resources, and persistence, which may vary widely: from low-skill, low-effort individuals to highly

capable and determined actors. For this reason, the estimation of the factor is measured as the percentile value representing the Threat Agent's relative position on the distribution of potential attackers. The Open FAIR risk analysis estimates the minimum, maximum, and most likely values of Threat Capability. These values reflect the expected range of attacker skill, from the lowest skill level expected to the highest possible one, along with the most likely level based on available information. It's important to note that, these assessments are specific to the threat vector in question, as an attacker may be proficient in one method of attack but lack skill in others. Since by definition, Resistance Strength is closely linked to the value of Threat Capability, this too must be measured as a percentile value. Once the ranges for Threat Capability and Resistance Strength are estimated, a Monte Carlo simulation is applied to compare random values from each distribution. The Vulnerability factor is then calculated as the probability that, in a given Threat Event, the Threat Capability surpasses the Resistance Strength, indicating a successful exploitation.

The other half of Figure 2.7 depicts the factors, and sub-factors, that consent to evaluate the Loss Magnitude (LM, also referred to as impact), meaning the "total financial value lost when a Loss Event occurs". It is always evaluated from the perspective of the Primary Stakeholder, the "person or organization that owns or is accountable for an Asset" and who experience the financial impact resulting from the Loss Event, and it is expressed as a multitude of losses, including productivity, response, replacement and reputation. Usually, productivity and replacement costs are considered as Primary Losses, while the others are commonly attributed to Secondary Losses. The Primary Loss Magnitude (PLM) represents the direct consequence of a Loss Event in terms of economic costs, and it can be estimated taking in consideration the best/worst case or the most likely scenario. The PLM value is then computed as the sum of the loss forms that are direct consequences of the Loss Event, as shown in the equation 2.4.

$$\text{PLM} = \sum(\text{Primary Loss Forms}) \tag{2.4}$$

Furthermore, the Loss Event can result in reactions from Secondary Stakeholders, "individuals or organizations that may be affected by events that occur to Assets outside of their control" and examples could be consumers of an organization if their personal private information are inappropriately disclosed. Their response may cause multiple additional losses for the Primary Stakeholder, generating the so called Secondary Losses. Each Secondary Loss has two primary components: Secondary Loss Event Frequency (SLEF) and Secondary Loss Magnitude (SLM). Secondary Loss Event Frequency refers to the estimated likelihood that a given scenario will produce secondary effects. Despite being termed a "frequency," it is actually a conditional probability, expressed as a percentage, indicating how often a Primary Loss is expected to lead to a Secondary Loss.

$$\text{SLEF} = Pr(\text{Secondary Loss} \mid \text{Primary Loss}) \tag{2.5}$$

Secondary Loss Magnitude represents "the losses that are expected to come from dealing with Secondary Stakeholder reactions", examples are fines and judgments,

loss of market share, reputation. Its evaluation is equal to the sum of all the loss forms resulting from the reactions of Secondary Stakeholders.

$$\text{SLM} = \sum(\text{Secondary Loss Forms}) \tag{2.6}$$

It is important to note that for a Secondary Loss to occur, there must have been a Primary Loss that caused a reaction from a Secondary Stakeholder.

The resulting Loss Magnitude is then obtained by the sum of the Primary and Secondary Loss Magnitudes

$$\text{LM} = (\text{Primary Loss Magnitude}) + (\text{Secondary Loss Magnitude}) \tag{2.7}$$

To further refine the estimation of both Primary and Secondary Losses, the model incorporates loss factors, which influence the magnitude of the impact based on characteristics of the asset, threat, organization, and external environment. Asset and threat related factors are typically categorized as Primary Loss Factors, and they recognize the volume and value of the Asset and the action and competence of the Threat Agent and whether they are internal or external to the organization. While organizational and external environment-related ones are considered Secondary Loss Factors and they are specific to the organization or the external legal and regulatory environment, respectively.

Finally at first glance it is possible to notice a substantial difference with the model proposed by Stallings: the role of Controls. Whereas in the diagram in Figure 2.2 controls were considered solely in the evaluation of likelihood, the current diagram does not present an explicit factor representing them. As Figure 2.8 shows, the Open FAIR risk analysis model incorporates controls in both the estimation of the Loss Event Frequency and the calculation of Loss Magnitude.

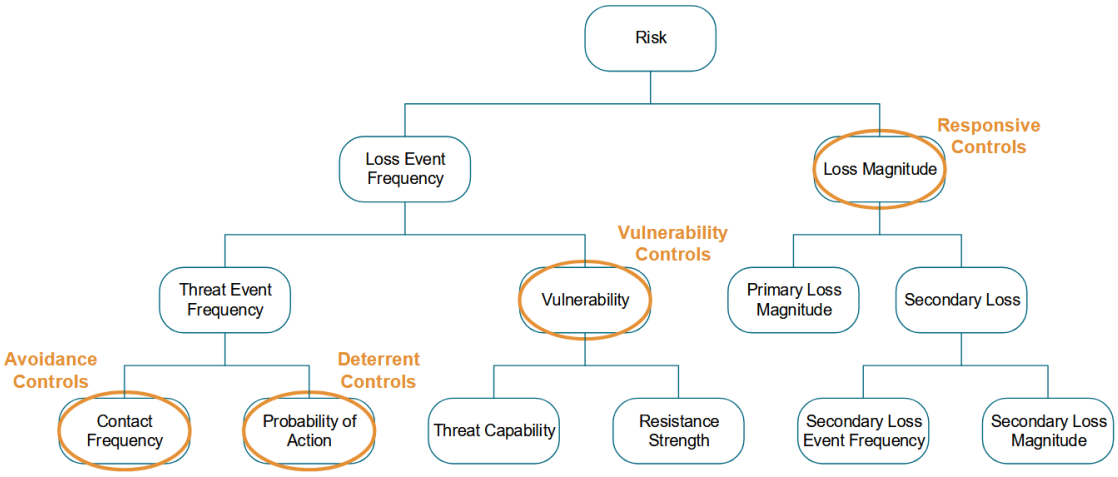

Figure 2.8. Open FAIR role of controls

For this reason an updated definition of Control should be "any person, policy, process or technology that has the potential to reduce the frequency and/or magnitude of future loss" [14].

The Open FAIR model has established 4 categories of Controls, based on how they affect risk:

1. Avoidance Controls, which affect the frequency or the probability that a contact between the Asset and the Threat Agent will be established. Examples of avoidance controls are firewall filters and physical barriers.

2. Deterrent Controls, which affect the probability that a Contact Event becomes a Threat Event. They are usually logging and monitoring systems.

3. Vulnerability Controls, which affect the probability that a Threat Event will result in a Loss Event (usually changing the Asset's Resistance Strength). Examples are authentication systems and access privilege mechanisms.

4. Responsive Controls, which affect the Loss Magnitude, limiting either the Primary Loss Magnitude or the Secondary Loss. Usually they include backup and restore processes and incident response capabilities.

The applied approximation techniques used by the quantitative method introduce inaccuracy into the FAIR model. To enhance the expressiveness and flexibility of risk modelling many studies have been conducted and one of the most promising is the one proposed in [16], where probabilistic techniques such as Bayesian networks can be employed to capture the conditional dependencies between events. In this context, the traditional Open FAIR framework is extended through the use of Bayesian graphs to model interrelated risk factors with greater precision. A notable example is the integration of FAIR into Bayesian networks for assessing Loss Event Frequency (LEF) in smart grid cybersecurity, as demonstrated in [17]. This approach uses structural analysis and conditional probability tables to make cyber threat analysis more detailed and easier to compare even with fuzzy inputs.

Chapter 3

Network Security Automation

In the context of network systems, the evolution carried forward by network softwarization, particularly through technologies as Network Function Virtualization (NFV) and Software-Defined Networking (SDN), has significantly enhanced the agility, resilience, and flexibility of modern network systems. Moreover, the increasing automation in the configuration of networks has made it more feasible to create complex, large-scale systems. In this scenario, deploying network updates has become simple and straightforward, making them extremely frequent [18]. At the same time, the security update speed for the entire system must keep pace with these changes. The high dynamism of modern computer networks puts a strain on the consistency of system security during updates, making this task even more challenging.

A security reconfiguration typically takes place when new security policies must be enforced to define or adjust the network's security behaviour. When a distributed security function is subject to a series of configuration changes, the preservation of its security throughout the entire process is questioned. In this scenario, a *transient* begins when the first configuration change is applied and ends when the last configuration change is implemented. In the several intermediate states, where the network configuration has been modified from the initial one but the final update has not yet been provided, an unspecified execution order may result in insecure stages [8]. An example could be removing a function instance before deploying the new one, resulting in a temporary vulnerability in the system. This problem turns out to be crucial for a particular type of network security function: the distributed packet filter firewall, the most common firewall technology used to enforce security policies. In this context, an incorrect scheduling of the configuration updates of the distributed firewall can lead to temporary service disruptions or, more critically, create temporary security breaches that attackers might exploit, thereby putting the connected systems at risk.

To face the open problem of optimizing firewall reconfiguration transients, new security automation approaches have been recently presented. One of most interesting solutions is the one proposed in *FATO*, which aims to define a scheduling of the configuration changes for a distributed virtual firewall that maximizes the number of secure states. The approach is exhaustively explained in [8] and this chapter presents the overall architecture of the tool and its modules, highlighting

the specific aspects to which this thesis has contributed.

3.1 FATO

FATO (FirewAll Transients Optimizer) is a framework that aims to combine automation, formal methods, and optimization strategies to overcome the problem stated above of reconfiguring transients. Optimality and high confidence in the correctness of the solution are guaranteed thanks to the formulation of a MaxSMT (Maximum Satisfiability Modulo Theories) problem, one of the most used approaches to pursue correctness-by-construction.

The problem faced by FATO is managing reconfiguration transients for a distributed packet filtering firewall. This issue implies two management aspects: establishing an allocation scheme and defining the filtering rules; both are typically created to enforce some connectivity policies, i.e., security properties applicable to communications that may occur within the network. The first one, the establishment of the allocation scheme, usually involves deciding the positions where the firewall instances should be allocated in the network service. Instead, the definition of filtering rules consists of generating rules through which each packet filter instance on the destination path can decide to forward or discard packets. When a transient is generated to update a new security configuration, at least one of the above-mentioned aspects differs from the initial layout. Some examples could be the need to establish a new firewall instance or the updating of the filtering rules to new connectivity policies. The number of intermediate states in a transient is equal to the number of modifications applied to the firewall configuration, such as deploying a new instance, removing the old one, and changing the filtering rules. Therefore, the firewall reconfiguration transient consists of a specific ordering of updating operations, including the additions, removals, and changes.

When the duration of a reconfiguration transient is significant, lasting more than a few seconds, it is crucial to maintain the security of connectivity policies. Suppose the transient lasts only a few seconds. In that case, an attacker typically does not have enough time to carry out access control violations, privilege escalations, or other types of attacks that could compromise these policies. However, in big virtual networks, where there can be hundreds of changes during each transient, the window of vulnerability increases substantially, and such extended periods provide perfect opportunities for attackers to exploit intermediate states in which services remain unprotected.

In light of these considerations, the objective of FATO is to "maximize the number of connectivity policies that are satisfied in each transient state" [8]. This solution leads to two primary outcomes:

- Each intermediate state will be as secure as possible, as most of the security requirements for reconfiguration are enforced within it.
- The security policies are satisfied as early as possible, with the primary goal being their enforcement from the initial states of the transient period.

Some policies are inevitably more urgent than others, and it becomes essential to establish an optimal schedule for ordering the firewall reconfiguration changes accordingly. This objective is efficiently achieved through the optimization process included in the methodology. Moreover, the FATO framework emphasizes another critical factor: formal verification. Ensuring a higher level of confidence in the correctness of the computed scheduling is an imperative requirement, especially in environments that involve safety-critical or mission-critical systems.

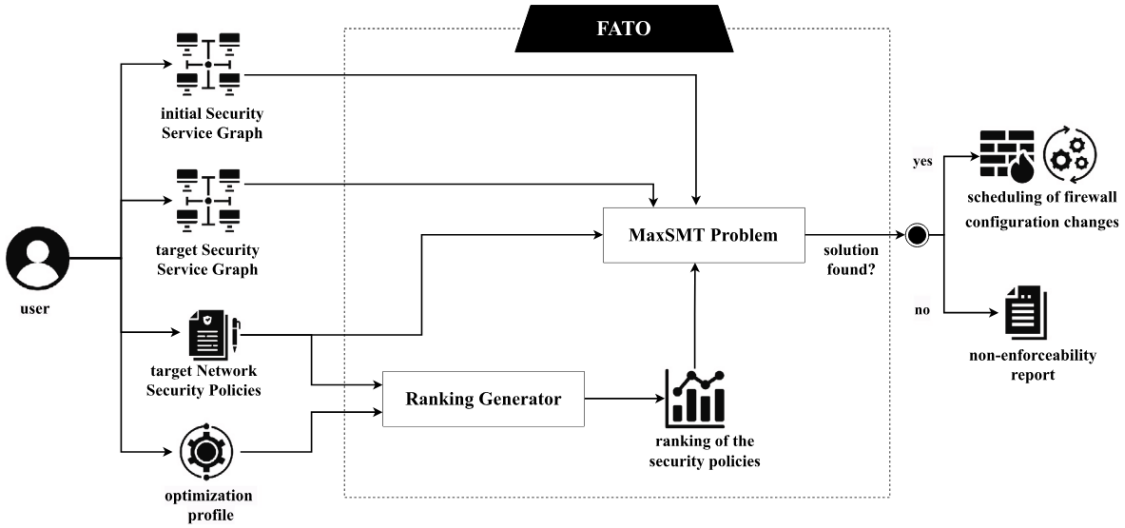

Figure 3.1. FATO approach workflow

Figure 3.1 illustrates the whole workflow of the FATO approach, showing the inputs specified by the user and the interaction among the different components. The four inputs needed by the framework are the following:

- the initial virtual network topology and the initial configuration of the distributed firewall;
- the target virtual network topology and the initial configuration of the distributed firewall;
- a set of target connectivity policies that must be satisfied by the target configuration. Optionally, a subset of policies, called *persistent policies*, can be added. This subset contains all the policies that must be satisfied throughout the whole transient, and not only in the final state;
- an optimization profile, which provides FATO the information necessary to establish the relative priority of the connectivity policies. Some of these profiles also need the specification of a partial or total order relationship for the policies.

The initial and target service graphs are referred to as *initial Security Service Graph* G_I and *target Security Service Graph* G_T, and they are represented as directed graphs. For each of them, the firewall configuration is composed of the

allocation scheme of the firewall instances and the filtering rules for each instance. The first one represents how the firewalls have been positioned in the network topology. Instead, the filtering set is composed of IP 5-tuple-based rules and a default action. The filtering rules belonging to the initial and final configurations are called respectively *initial Firewall Rules* R_I and *target Firewall Rules* R_T. Each firewall rule r specifies the condition $r.C$ that must be met based on the values of the five elements in the IP 5-tuple: source IP address, destination IP address, source port, destination port, and the transport-level protocol. It also outlines the corresponding action $r.a$ that must be put into practice when that condition is satisfied. Whenever a received packet does not match any rule in the filtering set, a default action is used.

From the union of G_I and G_T, FATO produces a new directed graph G_U, as shown in Figure 3.2. The new graph is referred to as *Union Security Service Graph*. It is used to consider all the possible paths in any situation, independently of when each node will be updated. In this graph, it is possible that two firewalls, with different indexes, refer to the same position. This is the case when the filtering rules for a firewall instance are changed between the old and the new configurations. Even though they share the same position in the allocation scheme and represent the same instance in the physical topology, the two firewalls are considered distinct entities in the model, as they differ in their filtering rules. An example is shown in the figure: over the link between the endpoint $e2$ and the routing instance $f7$, the two entities $p11$ and $f18$ cover the same allocation place.

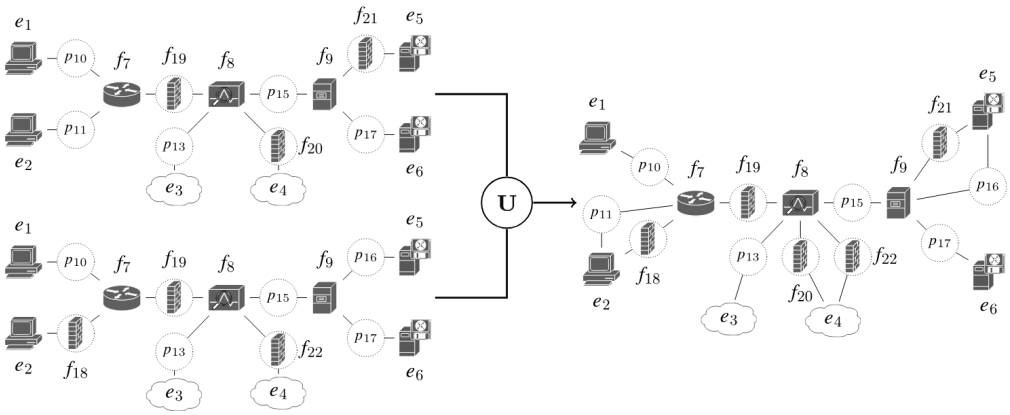

Figure 3.2. Union Security Service Graph

The set of connectivity policies *target Network Security Policies* P_T includes the policies that must be enforced in the updated configuration. They establish which packet flows must be blocked and which can reach the destination. Because of their nature, they can be divided into two subsets: isolation policies and reachability policies. The former aim to block incoming traffic, with *deny* as action to be applied. The others allow to define which flows are authorized to go through the network, with *allow* as action.

Finally, the user sends in input the *optimization profile* to indicate the relative priority of the connectivity policies. There can be different types of optimization profiles, and some of them might also require additional information about a partial

or total order relationship among the policies in the P_T set. Some of these have been defined in [8]:

- Security Max, which prioritizes the security of the system instead of the consistency of service, and it requires that the isolation policies have a higher priority than the reachability policies;
- Service Max, which requires that the reachability policies have higher priority than the isolation policies;
- Policy Max, which has the objective to maximize the number of policies satisfied in each intermediate state;
- State Max, which maximizes the number of intermediate states where each policy is satisfied, depending on an order relationship specified by the user. This is the only optimization profile that always requires a partial or total order relationship for the policies.

The first two profiles, Security Max and Service Max, allow the user to define a partial and total order relationship for each group of policies, i.e., isolation group or reachability group, specifying higher or lower priorities. It is important to note that an order relationship cannot be applied to persistent policies, as they must be satisfied in each stage from the beginning up to the end.

The FATO methodology begins by ranking all the policies received from the *target Network Security Policies* set, clearly excluding the persistent ones. This ranking is derived from the optimization profile and it considers also the partial and total order relationships. The process is computed in two phases:

1. the computation of the *dominance matrix* M_D, which represents the relationships between pairs of policies, in the non-persistent set;
2. the generation of the ranking based on the dominance matrix.

The dominance matrix is a square matrix with n rows and columns, where n is the number of non-persistent policies received in input by FATO. There can be two types of relationships inside the matrix: dominance and independence. The first one is represented by $p > p'$, meaning that the policy p has higher priority than the policy p'. The independence relationship indicates an equality in priority between the two policies, and it is represented by $p||p'$. Based on the prioritization profile, the dominance matrix can have different aspects. For the security-max profile, two constraints are applied:

- the isolation policies have higher priority than the reachability ones

$$\forall p|p.action = deny.\ \forall p'|p'.action = allow.\ p > p'$$

- policies with the same action are independent

$$\forall p, p'|p.action = p'.action.\ p||p'$$

For the service-max profile, the first constraint is the opposite. Instead, for the state-max profile, all policies with the same action are independent, since the objective is to maximize the number of states where each policy is satisfied.

Then, the computation of the dominance matrix is performed as follows: for each pair of policies p, p', the cell in the matrix $M_D[p, p']$ is one only if there is a dominance relationship between p and p', i.e., only if $p > p'$; otherwise, the value is zero. An example is shown in Figure 3.3, where the user decided to adopt a security-oriented profile.

M_D	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8
p_1	0	0	0	0	1	1	1	1
p_2	0	0	0	0	1	1	1	1
p_3	0	0	0	0	1	1	1	1
p_4	0	0	0	0	1	1	1	1
p_5	0	0	0	0	0	0	0	0
p_6	0	0	0	0	0	0	0	0
p_7	0	0	0	0	0	0	0	0
p_8	0	0	0	0	0	0	0	0

Figure 3.3. FATO Dominance Matrix

After the ranking has been computed, a MaxSMT problem is formulated from the initial and target service graphs G_I, G_T, along with their firewall configurations, the target policies and their ranking. The MaxSMT problem is a generalization of the SMT problem, which consist in determining the possibility to satisfy at the same time all the given first-order logic constraints. The main innovation is the introduction of the optimization objective and it is achieved by distinguishing two different sets of input constraints:

- *hard constraints*, that must be all satisfied in order to find a correct solution;
- *soft constraints*, which do not require to be satisfied in order to solve the problem. A weight is associated to each of them and the objective is to maximise the sum of weights associated to the constraints effectively fulfilled.

Hard constraints are used to model the input Network Security Requirements and all the other constraints imposed by the user, such as prohibiting the allocation of any service function in a specific allocation place. They are imposed on the *active* predicate, which represents the key element for the computation of the scheduling for the reconfiguration changes. This predicate is used to discriminate whether in a state s, a certain firewall instance or rule set is effectively in use.

For the intermediate states of a transient, three classes of hard clauses have been defined for the allocation and deletion of firewall instances:

1. for each node n active in both the initial and final stages, no state change is required in the transient. This means that, for each intermediate state s, the outcome of the *active* predicate applied to the node n is forced to be true.

2. for each node n such that its initial and final states are different, only one change of state is required in the transient. This means that, when applied to that node n, the active predicate changes value from a state s_i to the following state s_{i+1} only once. This constraint is enforced by imposing that the sum for the same node of all the differences for each pair of consecutive states is equal to 1.

3. only one operation is performed between two consecutive states, therefore each intermediate state is different from the previous only for one node at a time. In other words, given two consecutive states s_i and s_{i+1}, the outcome of the predicate *active* in the state s_i is different from the outcome of the predicate in the state s_{i+1} only for one node.

For the configuration updating in transient stages an hard constraint is also defined. As mentioned before, in this situation in the Union Service Graphs the same allocation place AP is shared by two nodes n, n'. When the node n is no longer active in a state s_i, the node n' becomes active in the state s_{i+1}. Therefore, even though the nodes are distinct in the Union Service Graph G_U, the outcome remains the same as if they were a single node, with only the configuration changed.

Moreover, the satisfiability of security policies can be translated into different classes of hard constraints. They are imposed over the *satisfied* predicate, which returns true if the given policy is satisfied in the considered state of the transient. It is important to note that all the elements of the set of Network Security Policies must be satisfied in the final state of the transient. In this context, for each policy, the role of hard constraints is to map the outcome of the *satisfied* predicate to the forwarding behaviour of the network functions in the paths crossed by the flows satisfying the policy's condition. There are two types of constraints, depending on whether they are applied to a reachability or isolation policy. In the first case, it is declared that in a state s the policy p is fulfilled if there exists at least a flow f satisfying the condition of the policy $p.C$, in other words if all the nodes of the paths crossed by that flow are active in that state and do not block its incoming traffic. On the contrary, for an isolation policy it is declared that in a state s the policy p is satisfied if for each flow f there exists at least a node of its path that is not active in that state, so it cannot receive traffic, or it is active and blocking the incoming traffic of f. Finally, for a persistent policy, the class of hard constraints imposes that the satisfied predicate must be true when applied in any state to a persistent policy, as that policy must be satisfied from the beginning to the end of the transient.

The soft constraints set is instead needed to formalize the optimization objective of the MaxSMT problem. Soft constraints are weighted to guide the MaxSMT solver in choosing clauses with the highest weight to maximise their sum. Weights are assigned starting from the policies with the lowest rank, derived from the output received by the Rank Generator. Policies with the same rank have the same weight. When a higher rank is achieved, the assigned weight is computed as the sum of all the previous weights plus 1. In this way, the soft constraints associated with policies of this rank will have a weight higher than the sum of all the weights of soft clauses associated with policies of a lower rank.

Finally, after the MaxSMT problem is built by combining the hard and soft constraints, a solver is employed to compute the optimal and correct solution. There are some cases where the solver cannot reach any solution for the problem. An case could be when the problem is not satisfiable, for example if a crucial allocation place cannot be used, or a persistent policy cannot be satisfied in all the intermediate states. In this case the solver assigns the most appropriate Boolean values for the *active* and *satisfied* predicates, in order to achieve most of the optimization objectives.

One aspect of the approach that has not been described in detail yet is where the *initial Service Graph* and the *target Service Graph*, and the respective firewall configurations, are derived from. They are both crucial elements for the correct functioning of FATO, but where are they taken from? The user has two possibilities: providing them manually or using two tools for their generation, VEREFOO and React-VEREFOO. An overview of the two tools is presented.

3.2 VEREFOO and React-VEREFOO

Security breaches and long re-configuration times are usually led by errors in manual configurations of security functions in computer networks. This problem is exacerbated for complex and large-scale modern networks based on network virtualization. To solve this issue, the two tools VEREFOO [4] and React-VEREFOO [7] have been proposed in literature. The first one defines a methodology to automatically define the allocation scheme and configuration of packet filters in the logical topology of a virtual network. Instead, React-VEREFOO focuses on an efficient method optimized for reconfiguring an already deployed virtual network.

3.2.1 VEREFOO

VEREFOO (VErified REFinement and Optimized Orchestration) is a framework that handles the creation, configuration, and orchestration of Network Security Functions in the logical topology of a virtual network, relying on three principles: automation, optimization, and formal correctness. VEREFOO manages the optimal allocation and configuration of the needed Network Security Functions by solving a MaxSMT problem. As already mentioned, the MaxSMT problem can be solved automatically, providing a formally correct configuration, that guarantees to satisfy all the hard logical constraints defined in the problem, while minimizing the number of used firewalls and configured firewall rules.

Similarly to the FATO approach, VEREFOO receives in input the virtual network's Service Graph and the target set of Network Security Requirements NSRs, as shown in figure 3.4.

A Service Graph is a logical topology characterized by a set of *Network Functions*, such as load balancers and traffic monitors, interconnected with network nodes, and that provide a comprehensive end-to-end network service. It is important to note that the Service Graph does not include security considerations, which are reported in the Network Security Requirements set. The main objective of a

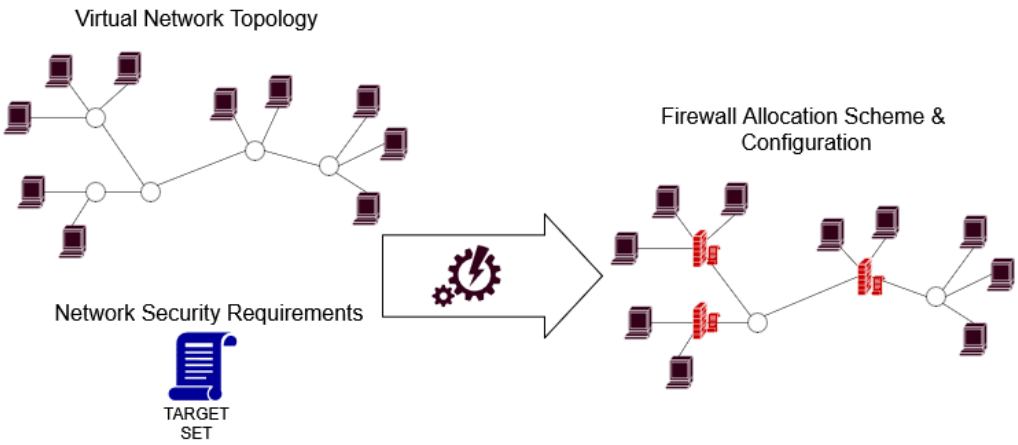

Figure 3.4. VEREFOO General Approach

Service Graph is to provide a more abstract view of the possible paths that packets can follow. For this reason, it is characterized by a complex architecture with multiple traffic paths from sources to destinations and even loops. In accordance with this concept, Network Functions are modelled considering only their ability to forward a traffic flow to its next destination, and the modification of its relevant parts, rather than each operation done by the function.

From the Service Graph, the VEREFOO framework automatically creates an *Allocation Graph*. The Allocation Graph is an internal representation characterized by the same set of Network Functions as the Service Graph, with additional nodes. These extra elements are placeholders called *Allocation Places APs*, generated for each link between pairs of network nodes or functions. Their purpose is to be possible places where the MaxSMT solver could establish if a Network Security Function needs to be allocated to achieve the optimal solution. After the computation of the MaxSMT problem, each Allocation Place will be marked as allocated or not allocate. The first case means that the specific AP has been chosen by the MaxSMT solver as optimal place. In the other case, the Allocation Place remains empty and it acquires a simple forwarding behaviour. More experienced users can have the possibility to either force the allocation of a firewall in a specific AP or prohibit its allocation in a specific AP. This flexibility of the VEREFOO methodology decreases the computation time by reducing the area where the tool must search for the solution. At the same time, it can lead to the impossibility of finding an optimal solution, as acceptable solutions might be discarded due to manually imposed constraints. Furthermore, if the user has a clear idea about the positions where to allocate the instances of the distributed firewalls, they can decide to immediately introduce an Allocation Graph to the framework, instead of the preliminary Service Graph.

The formal modelling of the Service Graph and Allocation Graph are the direct graphs $G_S = (N_S, L_S)$ and $G_A = (N_A, L_A)$, where N_S and N_A represent the sets of all the nodes in the respective graphs, while L_S and L_A are the set of edges, representing directed connections between nodes. The main difference between the two sets of nodes is that N_S includes only the endpoints and middleboxes, while N_A also includes the set of the APs where the firewalls can be potentially placed.

The second input is the set of Network Security Requirements, which must be satisfied by the firewall instances allocated on the Allocation Graph. Among all the network security properties, the VEREFOO methodology focuses on connectivity requirements between pairs of nodes, specifically reachability and isolation requirements. These respectively represent the need to allow or block a communication between two endpoints or subnetworks. As mentioned in FATO, each Network Security Requirement, independently of the type, is modelled specifying the IP 5-tuple of the allowed or prohibited flows. Therefore, they are formulated as combination of six elements: source and destination IP addresses, source and destination ports, transport-level protocol and rule type, chosen between reachability and isolation. The user must specify a general behaviour, which is the approach followed for the required security constraints. Each behaviour is characterized by a default action, defining how the framework should manage all the types of traffic for which no security requirement has been formulated. Four different approaches are provided:

- whitelisting, if all the communications for which specific requirements are not formulated should be blocked. The default behaviour is set to block all traffic flows and the user can only specify reachability requirements.
- blacklisting, if all the communications for which specific requirements are not formulated should be allowed. The default behaviour is set to allow all traffic flows and the user can only specify isolation requirements.
- rule-oriented specific, if there is no need to manually setting a default behaviour and the objective is only to minimize the number of rules.
- security-oriented specific, if the goal is to increase the security of the system, without choosing a default behaviour. In this case, only the communications that are strictly necessary in order to satisfy all user requirements are allowed.

Finally, after receiving the pre-processed Service Graph and the Network Security Requirements, the MaxSMT problem is solved. The expected outcome, if a solution is found, is composed by two elements: the allocation scheme of the distributed firewall instances in the network topology and the *Filtering Policy FP* set for each allocated firewall. The first one specifies the Allocation Places where the firewall instances have been allocated. Instead, the FP defines a set of filtering rules that constitute the configuration of the specific firewall instance. The FP consists of a default action, either whitelisting or blacklisting, along with an anomaly-free set of rules that specify how to manage particular types of traffic flows.

As mentioned in the FATO methodology, the MaxSMT problem distinguishes two classes of constraints. Hard constraints are used to model the Network Security Requirements and all the other constraints imposed by the user. Instead, soft constraints are defined to reach the two main optimization goals: to minimize the number of allocated firewalls and the number of rules for each allocated firewall. With the first one, the number of firewall instances necessary to enforce all Network Security Requirements is limited, thus minimizing the resource consumption. With the second constraint, the objective is to restrict the number of configured rules, thus reducing the memory required for storing them and enhancing the firewall's performance.

The positive aspects carried out by the solution of the MaxSMT problem, such as automation, optimization, and formal correctness, are a crucial step forward in automating network security. However, in order to make the approach scalable, the number and complexity of constraints in the MaxSMT problem must be limited. Therefore, its formulation is strictly linked to the modelling of network components and security requirements. Particular attention is given to the modelling packets in traffic flows. Each class of packets, also referred to as traffic flow t, is represented by a predicate defined over the values of the TCP/IP quintuple packet fields. Each traffic flow represents the behaviour of a specific packet class as it travels along a path, including how the traffic flow exits from the source node, how the intermediate nodes forward it, and also how the packet may be transformed as it passes through the nodes from the source to the destination. This means that all the packets represented by that specific traffic flow are forwarded and modified in the same way by the nodes along the path.

Not all the traffic flows present in the network are relevant to refining security properties. For this reason, only a subset of all traffic flows is considered. They are referred to as *interesting flows* and computed by processing the given set of Network Security Requirements and the configuration of the crossed nodes. To correctly model the relevant traffic flows, the VEREFOO methodology adopts the concept of *Maximal Flows*. Other approaches, such as the idea of Atomic Flows, have been taken into consideration and they have been exhaustively studied, implemented, and compared with the previous one in [19, 20]. Both approaches have their strengths and weaknesses: the computation time for generating Maximal Flows is less than that for Atomic Flows, but they require higher complexity for their representation inside the solver. The description and computation of the Maximal Flows approach have been described in detail in [4], so the following paragraph provides only a brief summary.

The objective of the Maximal Flows (MFs) approach is to reduce the number of generated flows, aggregating different sub-flows into a single one, smaller but equally representative for all the ones that have been joined. In order to do so, all the flows that behave in the same way while crossing the network are grouped in the same Maximal Flow. In this way, for the execution of the resolution algorithms of security management problems, it is sufficient to consider only the Maximal Flow and not each single flow that it represents.

Definition 3.2.1 *Called F the set of all possible flows of the network, the corresponding set of Maximal Flows F^M matches the following definition:*

$$F^M = \{f^M \in F | \nexists f \in F.(f \neq f^M \wedge f^M \subseteq f)\}$$

Therefore, the set F^M is a subset of F containing only flows which are not subflows of other flows. By applying this definition to the set of all possible flows satisfying the condition of a specific Network Security Requirement r, F_r^M groups all the flows behaving in the same way, and that need to be treated by the network in the same manner. Through this definition, multiple flows that behave similarly, by crossing the same nodes and receiving the same changes, are grouped into a single Maximal Flow. Considering only Maximal Flows helps reduce the number

of cases to be considered and the number of constraints composing the models. Moreover, since the generation of Maximal Flows occurs before the formulation of the MaxSMT problem, the number of free variables in the MaxSMT problem is limited, restricting the solution space and improving performance. For the resolution of the MaxSMT problem, all the Maximal Flows related to all the Network Security Requirements are computed by means of an algorithm that consider for each requirement the set of paths in the allocation graph that satisfy the condition of the requirement. Then for each of them, all the Maximal Flows containing it are added to the result set, iteratively.

3.2.2 React-VEREFOO

The VEREFOO methodology has been proven to be highly efficient for configuring a virtual network topology from scratch. However, it has resulted in being inefficient in those cases where the network already has some Network Security Functions installed and only requires reconfiguration. For these cases, an enhanced VEREFOO version, i.e., React-VEREFOO, has been proposed by with the objective of reducing the computation time for reconfiguration while still providing an automated, formally correct, and optimal placement and configuration of the required Network Security Functions. As already done for the description of the VEREFOO methodology, this chapter briefly summarizes the tool, while a detailed explanation is given in [7].

Similarly to the previous approach, React-VEREFOO receives in input the partially configured virtual representation of the network and the target set of Network Security Requirements, including the new requirements to be enforced in the updated network configuration. Additionally, it requires the initial set of NSRs, with the requirements already satisfied by the existing firewall configuration. The expected output is composed of the updated allocation scheme and the reconfigured filtering rules of the firewall, as shown in Figure 3.5.

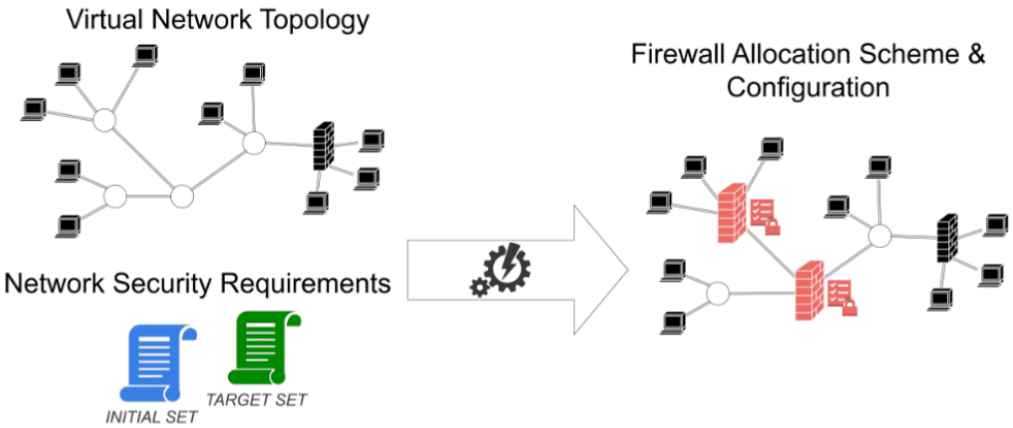

Figure 3.5. React-VEREFOO General Approach

After receiving those inputs, the React-VEREFOO approach is composed of multiple steps:

1. the definition of a complete and formal model representing the virtual network, the configuration of the network functions and the traffic exchanged;
2. the implementation of an algorithm able to detect the network areas that must be reconfigured by analysing the intersection of the two sets of Network Security Requirements received as input;
3. the resolution of a MaxSMT problem.

The primary step is very similar to the one presented in the VEREFOO approach, where the concepts of Service Graph, Allocation Graph have been presented and the modelling of the security requirements has been explained. However, a first discrepancy with the previous methodology is the use of *Atomic Flows (AF)* instead of *Maximal Flows*. The fundamental notion at the basis of the Atomic Flow approach is the concept of Atomic Predicate, described in [21]. For any given set of predicates, it is possible to compute a set of predicates that has been proven to be minimal and unique, such as the set of atomic predicates. Atomic Predicates have the following property: "The conjunction (disjunction) of two predicates can be computed as the intersection (union) of two sets of integers" [21]. In other words, it is possible to split each complex predicate into a set of simpler and minimal Atomic Predicates, and it can be represented by the set of integers identifying the Atomic Predicates. Then, from the idea of Atomic Predicates, the definition of Atomic Flows is derived. The objective is to divide each traffic flow into simpler, unique sub-flows so that it is possible to represent each predicate within the flow with its identifier. In this way, each traffic that can cross the network and each rule used to configure firewall instances can be expressed using only Atomic Predicates. The main advantage is that since predicates are inherently unique, it is possible to assign each one a distinct integer identifier; thus, the solver only operates with simple integers instead of more complex structures. As a result, the resolution performance of the solver is greatly improved. Furthermore, in the React-VEREFOO methodology, many operations are based on intersections and unions, which are computationally much less demanding when performed on sets of integers rather than on sets of complex predicates.

The central part of the approach is closely related to the contents of the two sets received as input. The Initial set of Network Security Requirements contains all the requirements already satisfied by the virtual network provided. Instead, the target set of NSRs comprises the new requirements that need to be enforced along with the old correct requirements that must be maintained in the new configuration. These requirements can be provided manually by the user, or by relying on policy extraction engines, as the one proposed in [22, 23], which automatically process IDS alerts and apply them to the network.

In this context, the intersection of the two sets, shown in Figure 3.6, produces three categories of requirements: Added, Kept, and Deleted. The first includes all the requirements present exclusively in the target set. On the other hand, the Deleted category comprehends all the NSRs present only in the initial set, and that are not needed in the final configuration. In the Kept category, it is possible to find all the requirements that are already enforced in the input virtual network and that must still be enforced in the final configuration. This general case might have

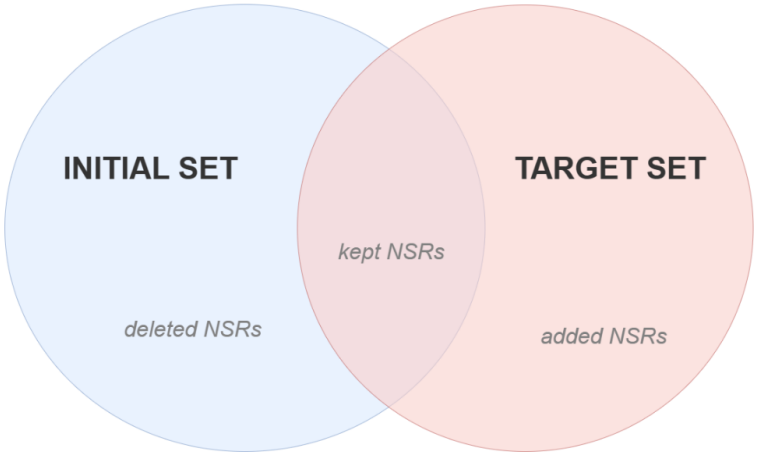

Figure 3.6. Intersection of Initial and Target Sets

two relevant extreme cases. The first one can happen if the two sets of initial and target NSRs are identical. In this scenario, the reconfiguration tool is useless, as no modification is needed. The other case can appear when the intersection of the two sets is empty. In this circumstance, reconfiguring the network is equivalent to a configuration from scratch, and the computational efficiency promised by React-VEREFOO may fail.

Once the three categories have been sorted, the designed algorithm detects the network areas that actually require firewall reconfiguration. To do so, it starts by analysing the elements of the network that should be modified because they are in conflict with at least one requirement in the set of added ones. At the end of the process, all the nodes that potentially fulfil the NSRs in the added group are selected as reconfigurable, as the algorithm is not able to decide a priori which would be the optimal path that must be allowed or the nodes crucial to block a specific traffic flow. It is important to note that, because of the different nature between isolation and reachability requirements, this part of the algorithm is formulated in separate ways. For isolation requirements, the procedure starts with computing all the Atomic Flows associated with each requirement. The algorithm verifies whether a node along the path is currently blocking the incoming traffic for that Atomic Flow. If the search produces no solution, all the nodes crossed on that path are selected as reconfigurable. Instead, for reachability requirements, for all the Atomic Flows passing through a specific path, the algorithm analyses whether the intermediate nodes are blocking the incoming traffic for that given flow. If at least one of such nodes is found, they are added to a temporary list and marked as reconfigurable.

Finally, the MaxSMT problem is formulated by joining the formal models of the inputs and the information about reconfigurable nodes, both for isolation and reachability requirements. Although the three principles of the problem, automation, optimization, and formal correctness, remain unchanged, some modifications to the hard and soft constraints have been adopted compared to the model proposed in VEREFOO. A first difference is that this approach restricts the set of Allocation Places available for firewall instances by keeping some of them as static

elements that the reconfiguration cannot update. The principle at the basis is that the optimal reconfiguration does not require updating the network, causing the lowest possible delay. In line with this principle, the soft constraints have also been modified so that an already installed firewall instance is preferred to a new instance to deploy, and to choose an already configured rule instead of generating a new one. To do so, the weights associated with deployed firewalls and configured rules are higher with respect to the weights for new firewalls and rules; in this way, using a reconfigured node would lead to a higher weights sum. As a result, the optimal configuration is the one that minimizes the overall resource consumption while also producing the fewest changes from the initial configuration. This last aspect also contributes to reducing the firewall reconfiguration transient management, thus providing support for the FATO methodology itself.

Chapter 4

Thesis Objective

This chapter presents the objectives of this thesis, framed in the context of the development of automated approaches for network security management.

As mentioned in the introduction, cyber threats become increasingly complex, involving multiple stages and diverse attack vectors. For this reason, modern networks face the increasing challenge of responding effectively and rapidly to security incidents. Existing network systems, especially those with large-scale architectures, often require the application of thousands of reconfiguration rules to fully restore security following an attack. In practice, this results in delayed responses, with reconfiguration times ranging from several minutes to much longer, depending on complexity, during which the network may remain exposed to unresolved or even new vulnerabilities.

This latency in response is particularly critical in the case of multi-phase or multi-vector attacks, where a single rule change may be insufficient to prevent further exploitation. Despite ongoing advancements in detection and monitoring, the research community has not fully addressed the problem of structured, automated reconfiguration of network systems in reaction to sophisticated attack scenarios. Most existing solutions focus on reconfiguring packet filter firewalls, while ensuring formal correctness, without considering a complete and timely realignment of the firewall policies in light of the overall risk context.

In such scenarios, the transition from the initial to the final configuration, with actions such as deploying or removing virtual functions or updating firewall rules, can temporarily leave the system in vulnerable states that lack the necessary protections. Even when short, these transitional states may expose the system to significant risks, including service disruptions, undetected intrusions, or the exploitation of temporary vulnerabilities by external actors. This issue is particularly concerning when the reconfiguration involves a sequence of dependent actions, each requiring non-negligible execution time.

To address this issue, the framework introduced in Chapter 3, called *FATO* (FirewAll Transients Optimizer), presents a first attempt to minimize the number of insecure intermediate states during firewall reconfiguration. Its goal is to schedule reconfiguration steps in a way that reduces the exposure time to threats. However, the FATO framework, in its original form, is not yet equipped to dynamically react to complex, multi-step cyberattacks based on a formalized evaluation of risk.

At the same time, structured frameworks for cybersecurity risk assessment provide valuable methodologies for identifying, evaluating, and prioritizing risks. Their primary focus is on long-term prevention and strategic planning. These models are highly effective in supporting organizations as they define security policies, assess vulnerabilities, and allocate resources according to threats' potential impact and likelihood. However, they are not inherently designed for timely and real-time responses.

In the face of rapidly evolving, multi-stage attacks, these frameworks fall short in their ability to react to evolving scenarios dynamically. This is mainly due to their dependence on historical data, predefined risk categories, and the need for manual interpretation of assessment outputs. Consequently, although risk models can inform what should be protected and why, they do not offer concrete mechanisms for how to act immediately when an incident occurs. The lack of a decision logic ready for automation limits their usefulness in time-sensitive environments where threats must be mitigated as soon as they are detected.

To address this limitation, this thesis proposes a novel approach that involves the readjustment of traditional risk assessment models to enable their use not only for strategic planning but also for guiding operational reactions to complex cyberattacks. Rather than treating risk analysis as a passive, periodic evaluation process, the proposed method recycles its outputs, such as calculated risk scores and threat prioritizations, as active decision drivers for automated defence mechanisms. In doing so, the thesis aims to bridge the gap between risk assessment and network security automation for cyberattack mitigation, setting the stage for more adaptive and resilient network security strategies.

This readaptation leads to the definition of a new semi-quantitative process, inspired by elements from the methodologies presented in Chapter 2, fundamental to understand the approach proposed in this thesis work. From the general model proposed by Stallings, it inherits the structured and process-oriented approach to risk identification and evaluation. From the Open FAIR framework, it draws on the formalization of factors like Loss Magnitude, Threat Event Frequency, and Vulnerability to quantify risk. However, instead of relying entirely on either framework, the proposed model simplifies and aligns selected aspects to support automation and real-time response. The semi-quantitative nature of the model allows the system to balance accuracy and computational efficiency, enabling fast prioritization of threats while maintaining a meaningful representation of risk.

The integration of this reactive layer is realized through its incorporation into the FATO framework. Within this extended version of FATO, the adapted risk assessment model allows risk-informed firewall reconfiguration strategies specific to each attack scenario. By linking the estimated severity and impact of ongoing attack patterns to reconfiguration priorities, the system is capable of responding more intelligently and efficiently to evolving threats. This allows the system to determine which reconfiguration steps should be executed first based on risk score, ensuring that the most critical protections are restored before others, thereby reducing the exposure to harm during the transition phase.

To validate the feasibility and effectiveness of the proposed solution, the FATO framework is tested against a realistic industrial scenario. The chosen use case

involves the reproduction of a real SCADA (Supervisory Control and Data Acquisition) system, representative of modern manufacturing environments. SCADA systems are exposed to security breaches due to their role in managing critical infrastructure, making them ideal for evaluating the capability of the system to respond quickly and reliably to complex cyberattacks. This evaluation demonstrates how integrating risk-based automation into firewall reconfiguration workflows can significantly enhance an organization's proficiency to rapidly and safely contain threats.

Chapter 5

Attack Risk Assessment Model

The approach proposed in this thesis aims to combine the main strengths of two fundamental aspects of information security: risk assessment and automatic network security. This research took inspiration from the world of risk assessment to build an effective tool for organizing responses to ongoing multi-vector or multi-stage attacks in computer networks, while ensuring that the evaluation of the risk is grounded in widely recognized principles. To achieve this goal, a new risk analysis model has been developed that combines methodologies and concepts taken from well-known risk assessment models. In this way, it has been possible to readjust the purpose of the risk analysis models to the high dynamism of the automatic reconfiguration of network systems, where decisions must be taken in a timely and justifiable manner. By adapting these principles to the dynamic requirements of automated security mechanisms, this approach makes it possible to translate risk scores directly into actionable priorities for network reconfiguration. The approach follows a semi-quantitative risk analysis in which numerical data are preferred to subjective evaluations. In this way it is possible to obtain objective and verifiable results. However, in the absence of precise data, numerical scales are associated with empirical judgments. This type of assessment combines the positive aspects of both quantitative and qualitative approaches proposed in Chapter 2, bringing together a strong scientific support for the obtained results with a simple communication for non-technical personnel. The result is a level of risk that can represent both damage and frequency for each attack, or stage in the event of multi-step attacks. By repeating the measurement for all attacks on the network it is possible to have a complete picture of the most dangerous attacks for the organization and then respond accordingly.

The new methodology is illustrated in Figure 5.1, and the following chapter explains in detail the notions and relationships between the different entities that are at the basis of the model. The approach is mainly divided into three parts: the calculation of the *Impact*, the assessment of the *Likelihood*, and finally the computation of the total *Risk*. A section of the chapter has been dedicated to each of these functions, highlighting not only the theoretical formulation of the factors but also their role in guiding automated reactions to cyberattacks. In particular, the *Impact* section describes how potential adverse effects are quantified in terms of economic and operational losses. The *Likelihood* section instead focuses on the probability of occurrence, analysing the conditions under which threats may be realized, with

reference to both the frequency of the attacks and the control effectiveness. Finally, the *Risk* computation integrates these two dimensions, producing a score reflecting the severity of the given scenario and that serves as a ranking score for prioritizing reconfiguration actions within the FATO framework.

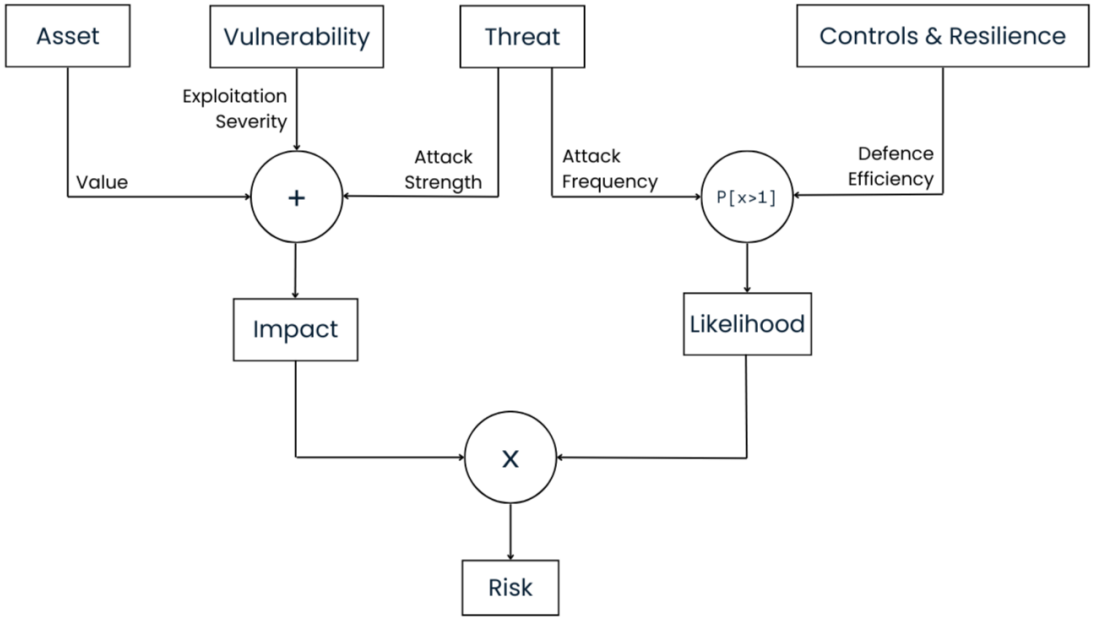

Figure 5.1. Attack Reaction Model

5.1 Impact

This first section focuses on the description of the factors that generate the *Impact* value and shows the new formula for their combination. Following the NIST definition, "the level of impact from a threat event is 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" [9]. For this reason, its value is closely linked to the economic, reputational, and operational damage that a specific attack can cause, as well as the severity of the vulnerability exploited by the attacker to damage the network. The factors that lead to the computation of the Impact score are mainly three:

- the *Asset Value*, which represents the value of any data, device, or other component of the environment that can be illicitly accessed, used, disclosed, destroyed, and/or stolen, resulting in an economic and reputational loss;
- the *Severity of the Exploitation of the Vulnerability* present on the node victim of the attack;

- the *Strength of the Attack*, meant as the sufferance of the node in terms of resource consumption.

In the model proposed in this thesis, the Asset Value can be integrated by the user in two ways: the first consists in the use of a parameter developed internally within the organization, called "Business Impact Score". The other way instead allows all organizations that, for various reasons, do not have this parameter available to estimate the value of the asset involved by referring to external sources such as annual breach reports.

In the first case, the Business Impact Score is a scoring system that evaluates the impact that different incidents or attacks have on the organization's performance. This parameter can be computed by means of a *Business Impact Analysis (BIA)*. The Business Impact Analysis is one of the core elements of a correct Business Continuity Management (BCM) plan, which aims to "keep on business service and operations during the occurrence of a disruptive event, IT-related, business-related, or a natural disaster" [24]. Business continuity management focuses on preventing, managing, and recovering from the consequences of attacks, incidents, or disruptive events. Its purpose is to ensure the continuous availability of the resources necessary to sustain critical business functions, so that they can continue without or with minimal disruption. Business continuity planning, as a component of operational risk management, defines the appropriate responses and the most cost-effective measures to adopt in the event of disruption, aiming to avoid interruptions to business activities. The process allows organizations to identify their key strategic vulnerabilities, priorities, critical resources, and functions, understanding the impact that a disruption of these activities can have on the organization, as well as ensure that they have plans ready to manage, maintain, and, in the event of a crisis, recover with minimal delay.

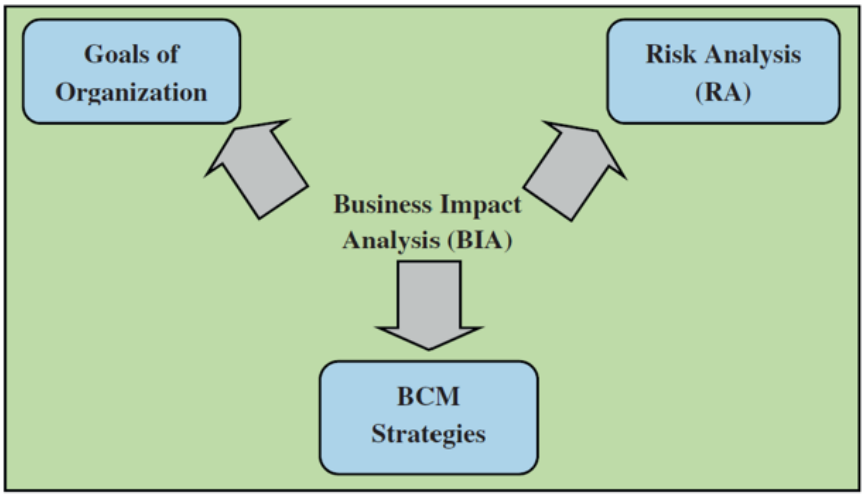

Figure 5.2. Business Impact Analysis

In the operational risk management scenario, therefore, Business Impact Analysis and Risk Assessment RA play crucial roles in an effective business continuity strategy. As shown in the Figure 5.2, the results of BIA and RA are merged to

generate correct Business Continuity plans. Moreover, to achieve the best results, there must be maximum alignment between these two areas and the organization's goals, both from a safety and economic point of view. As explained in [25], BIA is necessary to understand the organization, the processes, and how each department interlinks with the others in order to comprehend what the mission-critical activities of the business are. It returns a ranking of the organization's key products based on their identified critical functions and the computation of measures required to maintain their continuity, such as the Maximum Tolerable Period of Disruption (MTPD) and the Minimum Business Continuity Objective (MBCO) [26]. According to ISO 22301 (2019), MTPD represents the time necessary for undesirable impacts to become unbearable, and MBCO is the lowest acceptable operating level of each key product that allows an organisation to meet its business objectives during a disruption [27]. An example of a Business Impact Analysis measurement is the one proposed by [28], where the BIA is expressed as a composite variable taking into consideration all critical activities and resources (internal and external) required to maintain and resume the production of an enterprise's products. The factors included in the formulation are mainly four, and they are correlated to four hypotheses:

1. Staff numbers and expertise, which are linked to the assumption that "there is a statistically significant correlation between staff number, staffing level, and skills required to undertake organization's critical activities and the business impact analysis" [28];

2. Operational infrastructure, for which the hypothesis is that "there is a statistically significant correlation between premises, where the products are processed/manufactured/stored, including the manufacturing equipment used, and the business impact analysis" [28];

3. Technology and data communication. The idea at the basis is that there is a correlation between the technology used by the organization to perform critical activities and the BIA;

4. Supplies, for which the assumption is that supplies such as raw materials, energy, and services deeply affect the outcome of the Business Impact Analysis.

To support our choice to integrate the results of a Business Impact Analysis with the Risk Assessment, the researches carried out in [27, 29] also show the close link between the two disciplines. The first one is a framework that incorporates the BIA with a risk assessment matrix for critical physical assets and a mathematical model to estimate asset continuity parameters. Its model is then applied to a critical infrastructure as the petrochemical company. The other develops a framework that includes features that have not been incorporated in other integrations of BIA and risk assessment, such as capability assessments related to critical functions and information sharing between departments. This innovative approach has been evaluated in another critical infrastructure: a public sector organization. Finally, [26] integrates BIA and Risk assessment by developing a new framework in four steps: identify key products, break down their structures, define the associated

critical functions, and estimate the continuity parameters (i.e., Maximum Tolerable Period of Disruption and Minimum Business Continuity Objective).

Therefore, when an organization has the capability and resources to carry out a complete Business Impact Analysis, the results provide a highly accurate and context-specific representation of asset conditions and criticality. Such analysis is tailored to the organization's structure, processes, and objectives, ensuring that the derived Business Impact Score directly reflects the real operational dependencies and vulnerabilities of the enterprise. In this sense, the BIA values can be considered the most reliable input for risk assessment and continuity planning, as they capture the different shades of the business environment that cannot be generalized from external sources. Despite this, not all organizations, particularly small and medium-sized enterprises, may be able to conduct a complete Business Impact Analysis on their own. The reasons may range from a lack of expertise and resources to the absence of formalized risk management processes or even the high cost associated with detailed data collection and analysis. In such circumstances, an alternative approach becomes necessary. Organizations can seek help from external statistical information and empirical studies available in the field, such as annual breach reports and industry surveys. Although less precise, these external datasets provide crucial insights into the average impacts and common vulnerabilities observed across different organizations in the same field. By leveraging such data, organizations can approximate the value of their assets and the potential consequences of disruptions, thus enabling them to build at least a baseline model of their risk exposure. While these generic values do not achieve the precision of a Business Impact Analysis, they offer a practical solution for decision-making, especially when combined with expert judgment and contextual adjustments. The integration of external data ensures that, even in the absence of a formal BIA, organizations can prioritize their resources according to an empirical estimation of asset criticality.

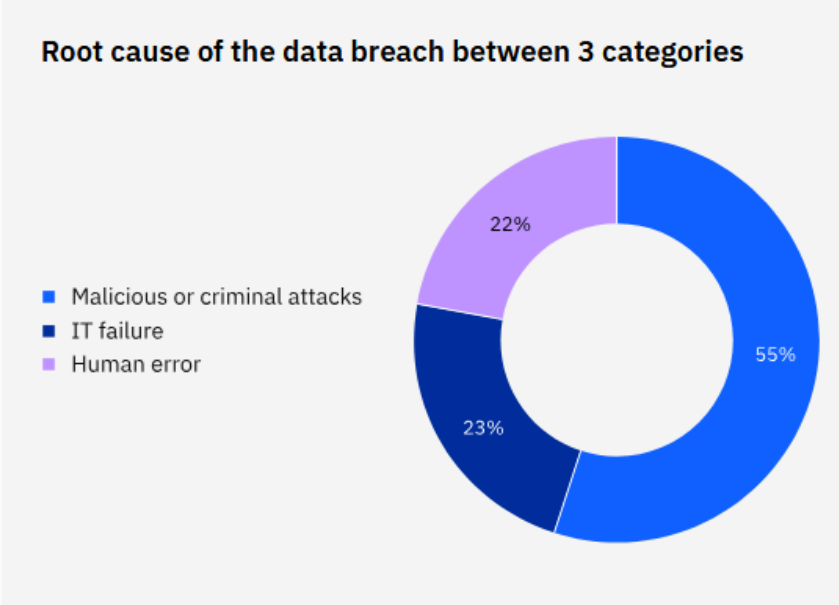

Figure 5.3. 2024 IBM Cost of a Data Breach Report

An example of a reliable and comprehensive external source is the annual *IBM Cost of Data Breach Report*. The IBM Cost of a Data Breach Report is one of the most widely recognized and comprehensive annual studies on the economic consequences of data breaches. It is based on an extensive dataset collected from hundreds of organizations all around the world that have experienced actual breach incidents. The report provides detailed statistical information on the average and median costs of data breaches, classified by industry sector, organizational size, geographic region, and other factors such as time to detect and contain the incident, security automation levels, or the presence of regulatory fines. By referring to the report's sector-specific data, an organization can approximate the likely financial consequences of security incidents affecting its assets. This makes it possible to integrate empirical, field-based evidence into the risk assessment process and to establish a semi-quantitative estimation of asset criticality. An application example could be the one extracted from the IBM Cost of a Data Breach Report 2024, in Figure 5.3, which reports the distribution of the primary root causes of data breaches. The diagram highlights how breaches typically originate from three primary sources: malicious or criminal attacks, human errors, and IT system failures, each contributing with different percentages to the overall occurrence.

In both cases, the values obtained from the formulation of the BIA and the data obtained from external sources must be reported on a scale from 1 to 10. In this way, all the parameters for calculating the Impact value are reported in the same range, thus avoiding imbalances in relevance.

The other two imperative factors for formulating the Impact value are the Severity of the Exploitation of Vulnerabilities and the Attack Strength. A vulnerability is defined as any "weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source" [11]. The severity of the exploitation represents how much damage the exploitation of a specific vulnerability in the system can cause. To calculate this parameter, the organization can rely on a well-known and internationally recognized system: the CVSS framework. The *Common Vulnerability Scoring System (CVSS)* is an open framework owned and managed by the FIRST.Org organization with the objective of communicating the characteristics and severity of software vulnerabilities [30]. This framework evaluates vulnerabilities through three distinct metric groups: Base, Temporal, and Environmental. The Base metric describes the inherent characteristics of a vulnerability, which remain unchanged over time, and assumes the reasonable worst-case impact across different deployed environments. It is divided into two subsets: Exploitability and Impact. The Exploitability metrics capture the smoothness and technical requirements for carrying out an attack, focusing on the characteristics of the vulnerable component itself. The Impact metrics, instead, describe the potential consequences of a successful exploit, reflecting the effects on the impacted component that suffers the damage. The Temporal metric captures factors that may evolve over time, such as the availability of exploits or patches. Finally, the Environmental metric reflects conditions that are specific to a particular user's system or operational context, including factors such as the presence of mitigations in that environment. Typically, only the Base score is published, and each organization is left to complete the data on Temporal and Environmental factors. The Base group produces a numerical score on a scale from

0 to 10, indicating the severity of a vulnerability relative to other vulnerabilities. This values can then be refined through the inclusion of the other two elements. To ensure clarity and consistency, each CVSS score is also expressed as a vector string, a concise textual notation of the selected metric values. For a better understanding, based on the value within the scale, the severity is also classified with qualitative parameters: low, medium, high and critical. It is noteworthy that the CVSS information does not include factors such as the number of customers, monetary losses due to a breach, or public sentiment on highly publicized vulnerabilities. Although these considerations may be relevant to an organization's vulnerability management process, for the purpose of this thesis, they can be excluded without loss of precision.

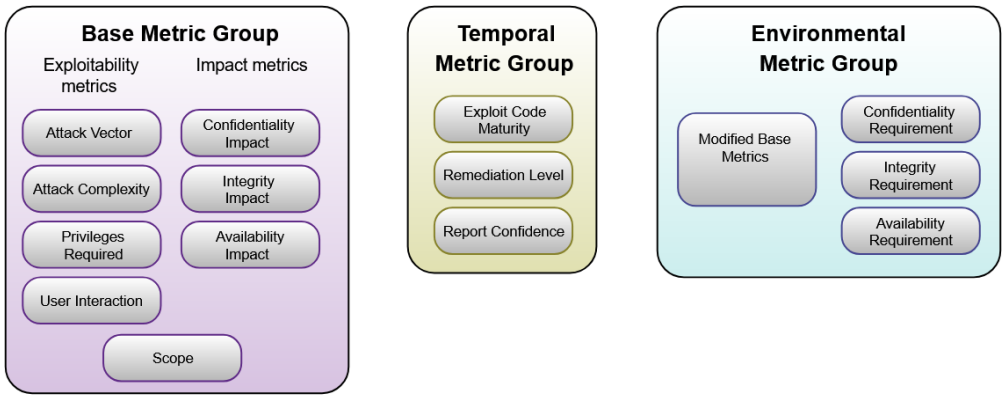

Figure 5.4. CVSS Metric Groups

To know the CVSS score of a vulnerability in the system, the organization must search for the CVE associated with that specific vulnerability. The *Common Vulnerabilities and Exposures (CVE)* system is maintained by The MITRE Corporation and is a "list of entries, each containing a unique identification number, a description, and at least one public reference — for publicly known cybersecurity vulnerabilities and exposures" [31]. Each entry is characterized by a CVE Identifier, an alphanumeric string that identifies a publicly disclosed vulnerability and it is composed of the "CVE" prefix, the year when the vulnerability was disclosed and some random digits. It is the organization's responsibility to keep an updated list of vulnerabilities in the system and the corresponding CVE IDs.

In cases where the CVSS score is not available, or when the corresponding CVE has not yet been identified, organizations can still rely on a qualitative, empirical evaluation to estimate the severity of a vulnerability. Considering elements such as the exposure of the component, the ease of exploitation, and the possible consequences of an attack, the outcome of this evaluation is expressed in terms of three qualitative categories: low, medium, and high. This classification provides a practical indication of severity even in the absence of numerical scoring. To ensure consistency with the rest of the model, these levels can then be mapped to symbolic values from 0 to 10 that allow the exploitation severity to be incorporated into the semi-quantitative framework proposed in this thesis. While less precise than a formal CVSS score, this approach still provides a structured and comparable way to

TABLE F-2: ASSESSMENT SCALE – VULNERABILITY SEVERITY

Qualitative Values	Semi-Quantitative Values		Description
Very High	96-100	10	The vulnerability is exposed and exploitable, and its exploitation could result in severe impacts. Relevant security control or other remediation is not implemented and not planned; or no security measure can be identified to remediate the vulnerability.
High	80-95	8	The vulnerability is of high concern, based on the exposure of the vulnerability and ease of exploitation and/or on the severity of impacts that could result from its exploitation. Relevant security control or other remediation is planned but not implemented; compensating controls are in place and at least minimally effective.
Moderate	21-79	5	The vulnerability is of moderate concern, based on the exposure of the vulnerability and ease of exploitation and/or on the severity of impacts that could result from its exploitation. Relevant security control or other remediation is partially implemented and somewhat effective.
Low	5-20	2	The vulnerability is of minor concern, but effectiveness of remediation could be improved. Relevant security control or other remediation is fully implemented and somewhat effective.
Very Low	0-4	0	The vulnerability is not of concern. Relevant security control or other remediation is fully implemented, assessed, and effective.

Figure 5.5. NIST Vulnerability Severity Table

integrate vulnerabilities into the risk analysis process. A useful example of a qualitative assessment is the one proposed by NIST with a pre-compiled table, shown in Figure 5.5, with which analysing vulnerabilities.

Finally, the Attack Strength Parameter represents the suffering of the node in terms of the percentage of CPU and memory used. To account for the severity of saturation, the model introduces a specific metric that evaluates the proportion of resources typically consumed during normal operations compared to the proportion exploited at the time of an attack. The reason is that many services or systems operate close to their performance limits under regular conditions, and even a modest increase in resource demand during an attack can trigger significant degradation or unavailability. An increase of usage of 20% from a daily proportion of 40% to the attack consumption of 60% is much less impactful with respect to the same increment between 80% and 100%. In order to express the degree to which the attack stresses the system beyond its ordinary workload, the formula (5.1) computes the Attack Strength parameter as a ratio between the percentage of resources typically used in daily operations and the percentage consumed during the incident. Therefore, higher ratios indicate a greater likelihood of service disruption, as the attack either consumes an excessive amount of resources or exploits conditions already near saturation.

$$\delta = \frac{attack - normal}{threshold - normal} \tag{5.1}$$

Since the value of the delta is mathematically a real number in the range from 0 to 1, in order to remain faithful to the scale chosen for the other two parameters, the result must necessarily be reported between 0 and 10.

Given the parameters described above, the Impact value is obtained by a weighted sum of the factors.

$$\begin{aligned}\text{Impact} = w_1 \cdot \text{Asset Value} + \\ w_2 \cdot \text{Vulnerability Severity} + w_3 \cdot \text{Attack Strength}\end{aligned} \tag{5.2}$$

Where the weights associated to each element are chosen by the user in order to establish relative importance among them. Each weight is a real number between 0 and 1, and the sum of all weights must be equal to 1, in other words

$$\sum_{i=1}^{3} w_i = 1 \tag{5.3}$$

In this way, the organization is able to establish an internal priority between the three factors based on the internal directives, while maintaining a balance between the elements. A hypothetical example could be the case in which there is an Asset Value equal to 7, a Vulnerability Severity equal to 5, and the Attack Strength equal to 3. Assume that the user considers the impediment caused by the attack to be of greater relevance than the other factors, choosing 0.2, 0.2, 0.6 as weights, respectively. In this case, the Impact value is 4.2. If, however, the Asset Value is considered to be of higher importance compared to the other two elements, associating 0.6, 0.2, 0.2 as weights respectively, then the result becomes 5.8. An extreme case can occur if a factor is equal to 1, but the associated weight is maximum, i.e., 1. In this case, the weights relating to the other two factors are zero, and the impact value is 1. On the opposite side, if all factors are equal to 10 and the associated weights are equivalent, i.e., all equal to 0.33, or if one element is equal to 10 and the others equal to 1, and the associated weights are respectively 1, 0, 0, the weighted sum returns the maximum value 10.

5.2 Likelihood

This second section focuses on the description of the factors that generate the Likelihood value and shows the new formula for its prediction. In the Open FAIR model, the Likelihood, referred to as LEF (Loss Event Frequency), reflects how many times the Loss Scenario is expected to occur in a given timeframe. A question that could arise is why should the probability of an attack happening again be considered when optimizing network reconfigurations? The reason is that some types of attacks may not be critical from an impact perspective, but be so simple that they can be repeated infinite times in a short interval. This type of attack not only undermines the stability of the system with small acts of sabotage but also ensures that the resolution algorithm never reaches a conclusion, as every few moments, the scenario it is working on is modified. The study carried out in this thesis to formulate the probability of reoccurrence focuses on two parameters:

- the *Attack Frequency*, which is the frequency with which an attack occurred in the past;
- the *Defence Efficiency*, which represents the capabilities of existing monitoring and defence systems.

In the risk analysis model defined by Open FAIR, the motivations that drive the attacker to act are also taken into consideration. These motivations, which may range from financial gain to political interests or the pursuit of notoriety, are considered an important parameter for understanding the likelihood that an attack will actually be carried out again in the future. However, in the model proposed in this research, this factor has not been included. The reason lies in the fact that, during the unfolding of a real cyberattack, it is almost impossible to identify the intent behind the adversary's actions accurately. Motivations are often hidden and ambiguous, and can only be hypothesized after long and detailed forensic analyses, which are not compatible with the responsiveness required for an automated reaction process. For this reason, while acknowledging the theoretical importance of adversarial motivation, this research has chosen to exclude it from the operational model, focusing instead on measurable and observable parameters that can be used in real time to drive the reconfiguration of network defences.

The Attack Frequency parameter reflects the number of times an attacker successfully exploited the threat event and performed the attack. As with Asset Value, also in this case the user has two ways of obtaining the requested data, from internal or external sources. In the first case, the organization can rely on historical records, databases, and updated reports regarding all incidents that have occurred over the years, with what success rate and how often. This is the best situation possible as the organization has the possibility of formulating precise and customized results of the components actually in use.

However, this scenario is not always feasible due to several practical limitations. First of all, not all organizations have the time and resources to collect and maintain a complete archive of past incidents. Small and medium enterprises may lack the tools or knowledge to monitor and document every attempted intrusion, resulting in inconsistent and misleading data. Moreover, newly created organizations do not have reports from past years on any incidents available. Even larger organizations may face difficulties in updating information across departments or over long periods of time, especially if the data has not been systematically recorded. In such situations, a practicable alternative is to rely on external sources of information. Annual reports from security vendors, industry-wide surveys, and government-issued studies provide valuable statistics on the most common attack vectors, their relative frequency, and their success rates. Even though these sources cannot represent the exact behaviour of cyberattacks within the organization's environment, they can provide a valuable guide for estimating the frequencies with which they may recur. In particular, they highlight recurring patterns of cyber threats within the same industry sector, offering practical hints about which attacks are more probable and how often they are likely to occur. Thus, even in the absence of a detailed internal dataset, the organization can base its assessment on solid empirical evidence and approximate its exposure to different attack scenarios with reasonable accuracy.

An example of a report based on industry-wide investigation is Verizon's annual "Data Breach Investigation Report". It provides a comprehensive examination of thousands of security incidents that actually happened to companies in different sectors, from medical to educational to manufacturing. Cyberattacks are classified both by category, such as social engineering, denial of service and privilege misuse, and by the type of organizations affected. For each attack, the frequency with

which it occurred during the year, the number of attacks that resulted in an actual data breach and optionally the type of data that was compromised are reported. An example is shown in Figure 5.6, where the information about "Miscellaneous Errors" are represented. As illustrated in the figure, 30% of incidents occurred in 2024 in this category originate from system misconfigurations. This finding further highlights the relevance of the topic carried out this research.

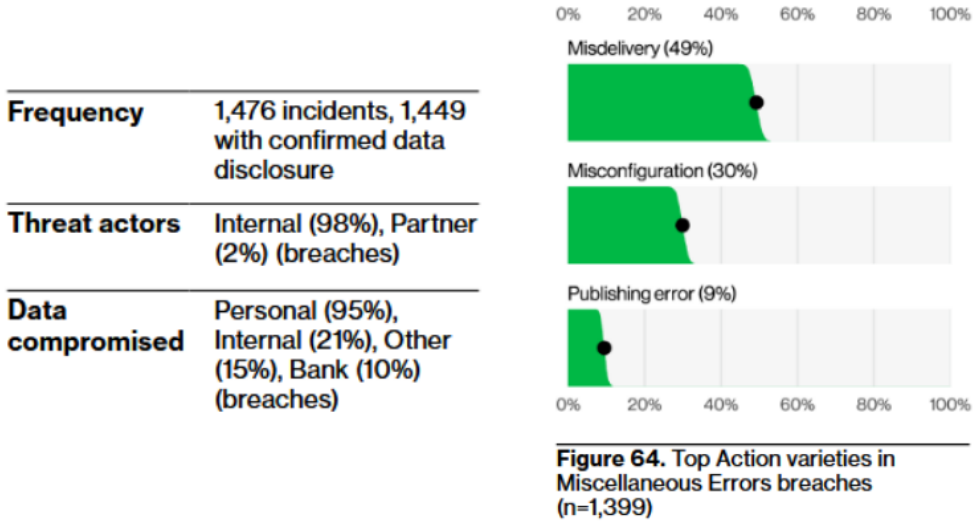

Figure 5.6. example of Attack Frequency from Verizon Report

The second parameter is the Defence Efficiency, which reflects how effective the monitoring and security mechanisms already deployed within the network are in limiting the impact of an attack. This parameter is introduced because the recurrence of attacks is strongly correlated with the level of damage that they are able to inflict. In fact, attackers are more likely to repeat or persist with attack strategies that always achieve the expected results. On the contrary, techniques that are easily mitigated by existing defences become less attractive to use again in the short term. For instance, if a particular Distributed Denial of Service attack is identified with a high detection rate and 60% of the malicious flows are blocked, then the likelihood that an attacker will repeatedly attempt the identical attack decreases significantly, as the probability of failure exceeds the potential benefits. On the contrary, attacks that encounter little resistance are more likely to reappear frequently, as their success rate justifies continued effort. The Defence Efficiency parameter introduces into the model the realistic assumption that adversaries will adapt their behaviour based on the perceived success of their actions, modifying the estimated frequency of occurrence over time.

Finally, the Likelihood value is the probability that the same attack will be executed in the future, based on the security measures already in place. To compute this value, the most suitable tool is the *Poisson distribution*. A Poisson distribution is a "discrete probability distribution that expresses the probability of a number

of events occurring in a fixed period of time if these events occur with a known average rate and independently of the time since the last event" [32]. Calling λ the known number of occurrences in a given interval, in other words the mean rate of the event, and k the exact number of occurrences for which we want to know the probability, then the Poisson equation is the following one:

$$f(\lambda; k) = P[X = k] = \frac{\lambda^k e^{-\lambda}}{k!} \tag{5.4}$$

where λ is a positive real number and k is a non-negative integer, usually $k =$ 0,1,2. As a function of k, this expression represents the probability mass function. In practice, it defines the probability distribution of a discrete random variable, specifying the likelihood that the variable takes on an exact value within its domain. The result is a mapping that connects each possible outcome of the random variable to its associated probability, ensuring that all probabilities lie between 0 and 1 and that their total sum equals one.

To adapt this formula to the calculation of the Likelihood, let's consider λ the constant average of the Attack Frequency. Limiting the analysis to the probability of an attack recurring based only on its past frequency does not allow taking into account security measures already in place. In fact, without the introduction of the defence efficiency parameter, the implicit assumption is that every attempted attack has the same chance of success, which is unrealistic. For this reason, the frequency of the attack must be reduced to the effective frequency of attacks that actually pose a risk, in other words, the ones that bypass the controls. To do so, the complement of the effectiveness of the controls, (1 − Defence Efficiency), is introduced and it allows to reflects the proportion of attacks that can avoid the implemented security systems. The final equation for calculating lambda is:

$$\lambda = \text{Attack Frequency} * (1 - \text{Defence Efficiency}) \tag{5.5}$$

Then, as we need to know the probability that the attack event will occur at least once in the interval of one year, the Poisson equation becomes as follows:

$$P[X \geq 1] = 1 - P[X = 0] = 1 - \frac{\lambda^0 e^{-\lambda}}{0!} = 1 - e^{-\lambda} \tag{5.6}$$

Because of the nature of the exponential value, the result is a real number in the range between 0 and 1. However, to remain consistent with the range of values assigned to the Impact value, the Likelihood must also be reported on a scale from 0 to 10, approximated to the second decimal place.

5.3 Risk

In the last part, the *Level of Risk* is computed as the product of the Impact and Likelihood values.

$$\text{Risk} = \text{Impact Value} \times \text{Likelihood Value} \tag{5.7}$$

The two factors of the operation are both limited in a range from 0 to 10; therefore, the risk levels obtained from their combination have a minimum value of 0 and a maximum value of 100. The result is a value that reflects both the economic and operational impact of the single attack, or of the single stage in the case of multi-step attacks, and the frequency with which this attack will recur in the future. Thus, at almost the same frequency, an attack that is critical from the point of view of system vulnerabilities will have a much greater level of risk than an attack with more minor effects. On the contrary, if there is an attack with a lower impact but very high frequency, this will have a higher level of risk than an effective but much less frequent attack.

Once the risk values for all active attacks on the system have been calculated, they are classified in descending order, starting from the scenario with the highest risk level. Given this list, the FATO framework reconfigures the network, starting from the most dangerous attacks and moving up to those of least importance. In this way, it is ensured that the system is in the best security situation in all intermediate states of the transient.

Chapter 6

Validation

The approach developed in this thesis has been applied and tested on a realistic case study based on an industrial SCADA (Supervisory Control and Data Acquisition) system. SCADA architectures are widely used in critical infrastructures such as energy, water, and manufacturing plants, where ensuring continuous availability and resilience against cyber threats is of primary importance. They are built for reliability but often lack of built-in security features to protect them from cyber-attacks. For this reason, they strongly rely on firewalls for protection [33]. Their complex structure, along with their critical nature, makes them an ideal candidate for evaluating the proposed methodology.

The validation is conducted by reproducing a real SCADA environment and injecting real-world vulnerabilities documented in the field in 2024. This setup provides a representative and concrete scenario in which to observe how the proposed risk-driven reconfiguration process can respond to attacks and prioritize remediation actions. In this way, the test not only assesses the theoretical validity of the model but also highlights its practical applicability to real industrial scenarios, where security breaches can result in severe operational, economic, and even safety consequences.

There are three main types of attacks simulated in this validation: Distributed Denial of Service (DDoS), data exfiltration, and privilege escalation. A DDoS is a cyberattack in which the attacker seeks to make a machine or network resource unavailable to its intended users by temporarily disrupting the services of the victim. Denial of service is typically accomplished by flooding the targeted machine or resource with superfluous requests in an attempt to overload systems and prevent some or all legitimate requests from being fulfilled. The range of attacks varies widely, spanning from inundating a server with millions of requests to slow its performance, overwhelming a server with a substantial amount of invalid data, to submitting requests with an illegitimate IP address. A data exfiltration attack, instead, is the result of a malicious action with the purpose of illicitly transferring information from an information system. Finally, in a privilege escalation attack, the intruder exploits a design flaw or a configuration oversight in a software application to gain elevated access to resources that are usually restricted.

The following section explains in detail how these attacks, by exploiting known vulnerabilities, can jeopardize the security of a critical system such as SCADA.

Furthermore, a brief description of the system structure is proposed to give a greater understanding of the issues.

It is worth clarifying that parameters such as Attack Strength and Defence Efficiency are inherently tied to the specific historical data and technical configurations of the organization under analysis. The first one should be directly measured on the nodes of the system, capturing how each attack consumes the victim's resources, such as CPU and memory, during real incidents. In the same way, the organization's Defence Efficiency should be recorded and evaluated based on its operational infrastructure, reflecting the actual capacity of monitoring and protection mechanisms to mitigate or block malicious activities. These measurements would provide the customized and reliable values, ensuring that the computed risk levels represent exactly the conditions of the system under investigation. However, gathering such information requires access to real operational data in a physical environment, which is not feasible within the scope of this thesis. For this reason, and to enable the validation of the proposed model, the values associated with these parameters have been estimated based on realistic assumptions drawn from public reports, and plausible hypotheses on similar systems. This approach allows the model to be tested in a representative scenario while acknowledging that more precise results could be achieved by applying it directly to a physical system with real-time data collection.

6.1 Use Case: SCADA system

SCADA systems improve the efficiency of Industrial Control Systems, and they provide better protection to the utilized equipment by giving valid identification and quick alerts to the observing stations [34]. The basic architecture is composed of sensors, responsible for detecting information from the physical environment, a Remote Terminal Unit or PLC, a Master Terminal Unit, also referred to as DNP Master, and a centralized database containing past alerts and communications. The Remote Terminal Unit RTU is responsible for collecting real-time data and information from sensors and forwarding them to the Master Terminal Unit. The DNP Master is the central monitoring station, and it is in charge of communications with the RTU and, therefore, the supervision and control of sensors and endpoints' equipment. Then the graphical interface, Human Machine Interface (HMI), authorizes operators to monitor the data acquired and processed by the SCADA system. Nowadays, many organizations allow their employees to access stations connected to HMI remotely. While this allows for greater flexibility and uninterrupted monitoring, it also creates new access points for malicious agents.

Modern industrial organizations, however, rarely rely solely on this basic architecture. To address increasing connectivity requirements and more complex operational needs, additional network segments and security layers are introduced. Common examples include a public Demilitarized Zone (DMZ), which separates internal control systems from external networks to safely manage remote access and data exchange, a dedicated Vendors' Zone that allows third-party suppliers to connect for maintenance or software updates under controlled conditions, and a Utility Partners' Zone designed for secure interaction with external service providers

or cooperating infrastructures responsible for integrating the SCADA system with the rest of the industrial process. These additional components increase operational flexibility and interoperability, but they also introduce new potential attack surfaces. In this structure three main types of data flows are transmitted [35]:

1. telemetry data requests from the Master Unit to the Remote Unit;
2. data from the MTU to corporate or other DMZs;
3. data from the Master Unit to Balancing Authority and Utility Partners.

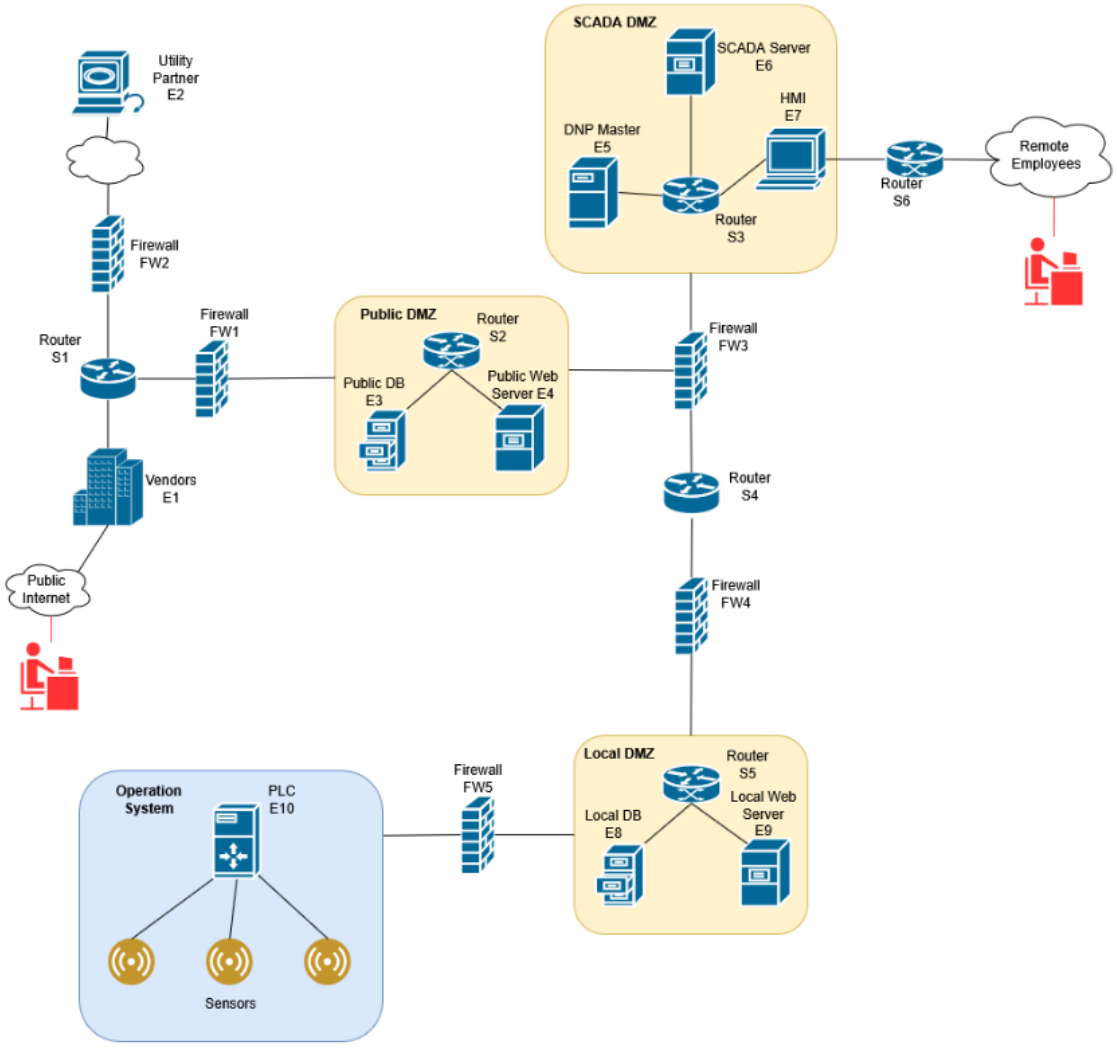

Figure 6.1. example of SCADA system

The specific SCADA network topology that was considered for the validation of the proposed approach is the one depicted in Figure 6.1. There, two possible attack points were considered. The first concerns an attack vector external to the SCADA environment, where the access credentials or communication channels of users linked to the Vendors' Zone are compromised and malicious actors gain entry

from outside the perimeter. This scenario reflects the risks associated with third-party access and supply chain connections, which often represent a weak point in industrial networks. The second scenario, on the other hand, focuses on an attack originating from inside the system, hypothesizing a successful infiltration into the remote communications with the HMI. This could occur through compromised maintenance accounts, misconfigured VPNs, or infected devices used by authorized operators. By leveraging these two entry points, the attacker is able to disrupt normal operations either by penetrating the network from the outside or by exploiting trusted internal channels, demonstrating how both external and internal vectors can threaten the integrity of a SCADA system.

Assuming there are no strict controls on traffic coming from a node in the public internet to the Vendors' Zone, an attacker is able to exploit this access to perform a multiple attack by taking advantage of known vulnerabilities of two components of the architecture: the Utility Partner and the Public Database. In the first case, it carries out a Distributed Denial of Service attack to block communications with the SCADA system. The second uses design flaws to perform data exfiltration over information and data relating to the SCADA system and the rest of the industrial process. As shown in Figure 6.2, the attacker exploits two principal vulnerabilities: the CVE-2024-7587 for the Utility Partner component and the CVE-2024-8309 for the Public Database. The first one puts the victim node at risk of DoS attacks. In fact, its description reports that an attacker, with valid local access to the system, can exploit a weakness caused by incorrect default permissions in the GenBroker32 component. This vulnerability affects systems where GenBroker32 is installed together with ICONICS GENESIS64 (version 10.97.3 and earlier), Mitsubishi Electric GENESIS64 (version 10.97.3 and earlier), or Mitsubishi Electric MC Works64 (all versions). By taking advantage of a folder configured with overly permissive access rights, the attacker can read or alter sensitive information stored by these products or even trigger a DDoS condition [36]. In other words, the attacker can tamper with data or disrupt the normal functioning of the software simply by accessing an improperly protected directory on the same machine. According to the Common Vulnerabilities and Exposures Database, CVE-2024-7587 has been assigned a CVSS Base Score of 7.8, which places it in the "high" severity category. This score reflects both the ease with which the vulnerability can be exploited and the potential impact on the affected systems. In the context of the Utility Partner component, an attacker who successfully exploits this weakness could launch a Distributed Denial of Service attack that disrupts its regular operation. By gaining control of the insecure directory, the attacker can overload or corrupt critical files used by the service, causing it to become unstable or unresponsive. As a result, legitimate communications between the Utility Partner and the SCADA system could be delayed or completely blocked, reducing the availability of critical data and undermining the reliability of the industrial process. This means that essential monitoring or control actions may be interrupted, creating a window of vulnerability in which the overall system performance and safety could be compromised.

The other vulnerability allows to perform SQL injection attacks through prompt injection if the specific port associated with the LangChain, port TCP 8000, has

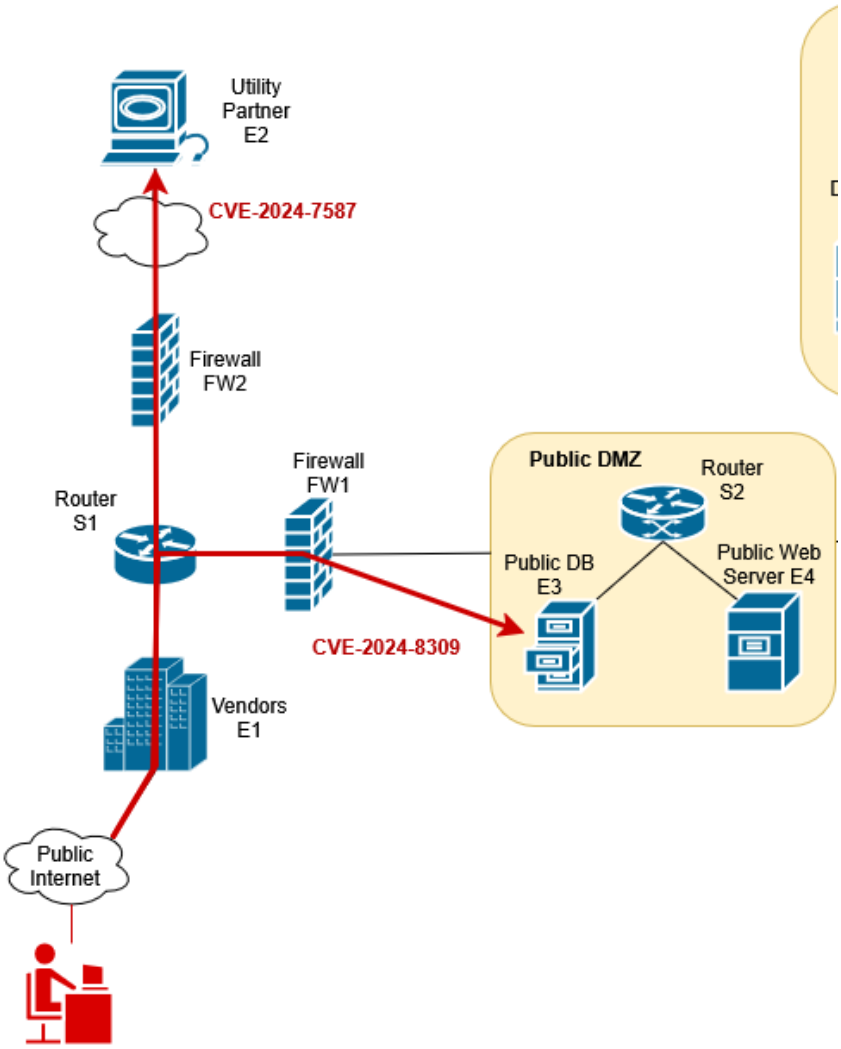

Figure 6.2. Attacks to external components

not been correctly filtered or closed. If exploited, this vulnerability can allow unauthorized manipulation of data, data exfiltration, or even deletion of critical information. This flaw may also lead to breaches in multi-tenant security environments and compromise the integrity of stored information [37]. In practice, an attacker can create, update, or delete nodes and relationships without proper authorization, extract sensitive data, and disrupt services across different tenants. According to the Common Vulnerabilities and Exposures Database, CVE-2024-8309 has been assigned a CVSS Base Score of 4.9, which places it in the "medium" severity category. Although not classified as high, this score still indicates a vulnerability capable of producing significant damage when exploited in the right context. In the case of the SCADA architecture, an attacker who takes advantage of this flaw to perform data exfiltration on the Public Database could gain unauthorized access to operational or configuration information shared across the Vendors' Zone and the Utility Partners' Zone. This type of information may include process parameters, maintenance schedules, credentials, or other sensitive records needed for the coordination of the industrial process. By extracting these data, the attacker could compromise

the confidentiality of the entire system and even prepare subsequent attacks with greater precision. In other words, unauthorized access to the Public Database may result in a loss of trust between partners, a breach of regulatory requirements, and an exposure of proprietary information. Furthermore, once this knowledge is in the attacker's possession, it can be used to disrupt services, manipulate configurations, or undermine the integrity of both the SCADA system and its connected components, amplifying the overall impact of the intrusion.

On the other side, assuming that the remote access to the HMI interface is not appropriately protected, an attacker capable of intercepting the communication channel may be able to get inside the SCADA system and disrupt its proper functioning. Once inside the system, the intruder could compromise the communications between the DNP Master and the PLC Remote Unit, disrupting the communication of control commands and sensors data that are essential for the real-time operation of the system. By interrupting or forging these exchanges, the attacker can cause delays, disseminate false alarms, or prevent legitimate control signals from reaching the security administrators. In addition, the attacker could alter or delete the data stored in the local database that feeds the HMI, leading to corrupted historical records, inaccurate process information, or the loss of configuration parameters. These actions would undermine the reliability of the SCADA environment, making it difficult for operators to distinguish between regular and malicious events, and potentially resulting in unsafe or unstable operation of the industrial process. The vulnerability that the attacker can exploit to reach their objective is the CVE-2024-10313, as shown in Figure 6.3 and it allows malicious agents to take advantage of a flaw in the way the software on the HMI node handles project template files. Following the description provided by the CVE Database, this vulnerability present on the iniNet SpiderControl HMI Editor may allow overwriting critical system files, potentially causing system paralysis, or placing files in startup locations, which can lead to remote control of the affected machine. In fact, if the software loads a malicious project template crafted by the attacker, it can be tricked into writing files to arbitrary directories on the system [38]. The CVSS Base Score associated is 8.0, which places it in the "high" severity category. The effects of the exploitation of such vulnerability are severe and debilitating both for the operation and security of the system. By overwriting critical system files, the attacker could corrupt configuration data, modify sensors' thresholds, or disable essential services, making key functions of the SCADA system inoperable. Moreover, malicious agents could leverage the victim node as a starting point to perform lateral movement actions within the network, gaining access to other sensitive components and expanding the scope of the attack. This would also allow the intruder to gain persistent remote control over the compromised node, enabling them to execute commands or deploy additional malware every time the system restarts. This persistence could be used to shut down or misconfigure safety mechanisms and introduce false data into the monitoring interface to conceal ongoing manipulation. Beyond the technical damage, the consequences related to this vulnerability may also lead to loss of stakeholder trust and significant financial and reputational harm for the organization.

Consequently, the Exploitation Severity values associated with each malicious flow, where by malicious flow we mean the data flow from a malicious source to a

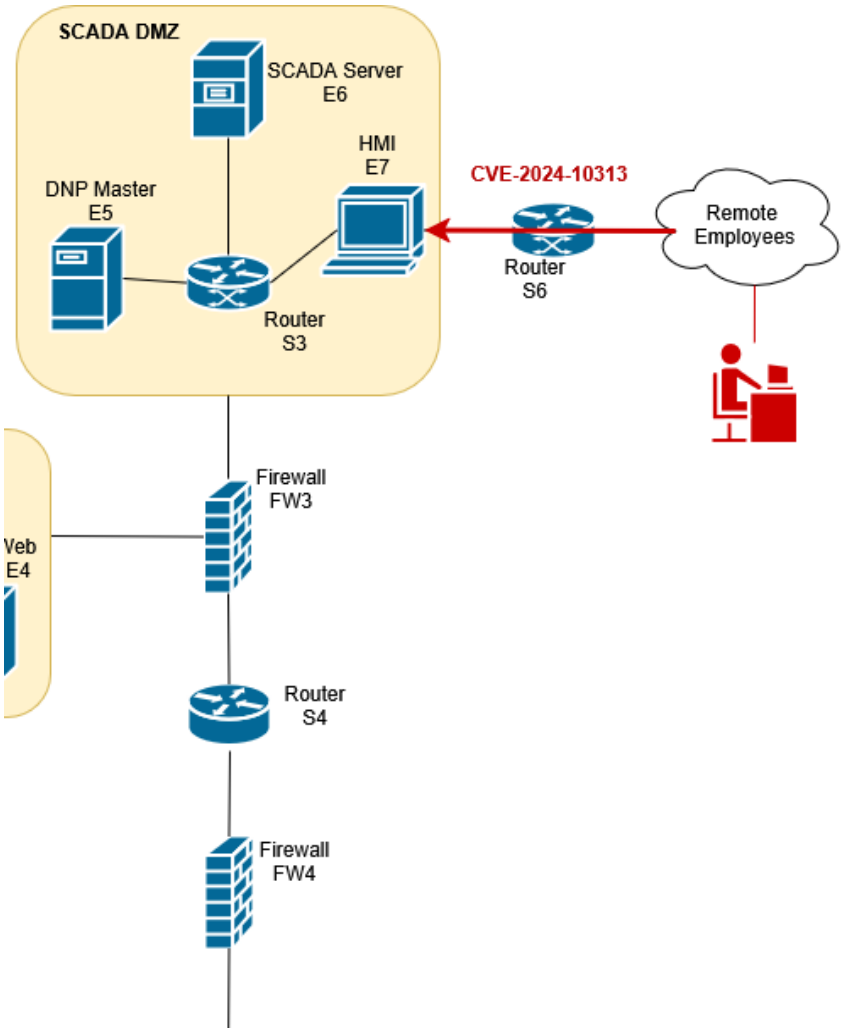

Figure 6.3. Attacks to internal components

victim node, are 7.8, 4.9 and 8.0 respectively.

As already mentioned in the description of vulnerabilities associated with each attack, each victim node has a different worth for the reputation and operability of the organization. In particular, the compromise of the Utility Partner's Zone can lead to medium to major damages from an operational perspective, as it directly disrupts the communications between the SCADA system and the rest of the process. Such a disruption affects the timely exchange of control commands and operational data, delaying or even blocking critical activities required for the stability of the industrial process. According to the IBM Cost of Data Breaches 2024 report, the average cost of known unpatched vulnerabilities of this kind is estimated at 4.33 million dollars. For this reason, and considering the potential cascading impact on the entire infrastructure, the score assigned to the asset value for this component is set at 4, reflecting its high criticality within the organization's operations.

On the other hand, the corruption of historical data on the Public Database leads to much less severe consequences. This database, although connected to the SCADA environment, does not directly influence the real-time operations of the

system or the organization's core processes. Its primary function is to store and share information about past alerts and communications that are accessible to Vendors and Utility Partners for coordination purposes. Therefore, any alteration or deletion of its contents, such as historical records and breach reports, does not immediately interrupt the control of the peripheral equipment or the execution of essential functions. Despite this, the reputational damage in the event that the incident becomes public knowledge is a relevant aspect to take into consideration, especially for organizations in critical fields. For this reason, according to the IBM Cost of Data Breaches 2024 report, the average cost of a data exfiltration incident is estimated at 5.21 million dollars. However, because the Public Database in this scenario is designed to contain only non-sensitive information and excludes operationally critical or confidential data, the potential financial and reputational impact of an attack is significantly lower. Therefore, considering the limited influence of this component on the organization's operational continuity, the Asset Value for the Public Database is set to 5, reflecting its moderate criticality within the broad infrastructure.

Finally, the compromise of the HMI interface is the most dangerous attack, both from an operational and reputational point of view. This is because the HMI provides direct access to the internal communications between the DNP Master and the Terminal Units, acting as the interface for the administrators through which commands and sensors information appear in real-time. An attacker who gains control over this interface can intercept or manipulate these exchanges, effectively taking control of the system's core. From a safety perspective, for example, this means that critical alarms raised by sensors in the physical environment might never reach the central control, delaying or preventing the response to unsafe conditions. Such a failure could result in equipment damage and environmental hazards, or even threaten human operators. From an operational point of view, an attacker could alter or delete configuration files stored locally on the HMI, leading to corrupted or missing settings, false displays, and the loss of essential parameters needed to run the industrial process correctly. These actions would make it extremely difficult for operators to trust the information they receive, severely undermining both the awareness of the actual situation and the decision-making in critical circumstances. In addition, a compromise of this level could have significant reputational consequences, as it highlights that attackers have reached the most sensitive part of the infrastructure. For these reasons, according to the IBM Cost of Data Breaches 2024 report, the cost of such highly disruptive attacks is roughly 5.7 million dollars. Therefore, the asset value assigned to the HMI interface is set to 6, reflecting its critical importance within the SCADA environment.

In conclusion, for all the motivations explained above, the Asset Values associated with each attack flow are respectively 4, 5 and 6.

The last parameter to consider is the Attack Strength. Due to the fact that the model studied could not be applied to a physical computer network, the values assigned to the malicious flows do not have empirical support. However, they are the result of reasoning about the nature of the attacks performed.

The first attack path considered is the one directed to the Utility Partner, where a Distributed Denial of Service is performed. In case of a DDoS attack, the resources

of the victim node are exhausted in order to degrade or block the availability of services [39]. To do so, the attacker needs to perform a massive campaign, often coordinated through Botnets, of fake requests that are able to consume the capability of the victim node. This type of activity overwhelms processing power, memory, and network bandwidth, preventing legitimate traffic from reaching the intended services. In an industrial setting, such an attack can severely disrupt the regular exchange of operational data between the SCADA system and external partners, delaying control actions or completely blocking them. The result is an extreme consumption of resources for the Utility Partner's endpoint, with an immediate impact on its ability to function correctly. Consequently, the assumed value for the Attack Strength is set to enter in the "high" category, with a value equal to 9.

The other attack performed on the public side of the system is the data collection and exfiltration carried out over the Public Database. During such an attack, the intruder needs first to collect the data that they consider relevant and then extract it. This often involves reconnaissance activities to identify which tables or records contain valuable information before initiating the transfer of the selected data. Since the database is public and does not hold sensitive operational parameters, the attacker does not need to perform large numbers of complex queries or overwhelming transactions. To do so, little to no requests are sent to the database, limiting resources consumption and avoiding detection mechanisms based on unexpected traffic volumes. In other words, this means that the attack can proceed slowly and undercover, causing minimal stress on system resources while still compromising the confidentiality of non-critical information. Therefore, the Attack Strength value related to the attack on the Public Database falls within the "low" category, with a quantitative value set to 3.

Finally, the attack performed on the internal side of the system exploits a vulnerability in the HMI interface to provoke a privilege escalation. This kind of attack occurs when the adversary takes advantage of a programming error in a program, service, or within the operating system software or kernel itself to execute adversary-controlled code [40]. Once the attacker gains this elevated level of access, they can bypass normal security restrictions and interact directly with critical functions of the SCADA environment. It is in the interest of the intruder to stay hidden as much as possible in order to operate undetected, often using stealthy techniques to avoid triggering alerts or anomalous usage patterns. For this reason, the resource consumption is limited as much as possible; in this way, the adversary can quietly alter configurations, extract sensitive data, or manipulate internal communications without drawing attention from security administrators. Therefore, also for this attack flow, the value associated with the strength of the attack is categorized as "low", with the numerical value equal to 3, reflecting the limited stress to which resources are voluntarily subjected.

In summary, the Attack Strength parameter for the three attacks is respectively 9,3,3. The final results for all the parameters for each attack vector are shown in the table 6.1.

Assuming that, for all the received attacks, the organization has no internal priorities between the three parameters, thus the three weights are associated with the same value of 0.33. In this case the respective impacts of the performed attacks

Summary			
Victim	Asset Value	Exploitation Severity	Attack Strength
Utility Partner	4	7.8	9
Public Database	5	4.9	3
HMI	6	8.1	3

Table 6.1. Impact Parameters Summary

are as follows:

- for the attack focused on the Utility Partner, the impact is 6.9;
- for the impact correlated to the Public Database, the impact is 4.3;
- for the attack oriented to the HMI, the impact is 5.7

Therefore, assuming only to consider the impact of each attack and not the full risk value, the reconfiguration order proposed to FATO should start with the update of the policies in the instance present between the Vendors' Zone and the Utility Partner's Zone. Then, it would require the deployment or update of a packet filter to protect the HMI interface, and it ends with the reconfiguration of the policies in the packet filter that protects the Public Database. In case the organization chooses a different configuration of parameters for each attack, the scenario may change. A first example may be a change in the selection of weights for the privilege escalation attack to the HMI interface, while maintaining the other as in the example above, assuming choosing the following weights: 0.7 for the Asset Value, 0.2 for the Exploitation Severity, and 0.1 for the Attack Strength; the Impact value is 6.1, which is still lower than the initial Impact for the DDoS attack, which was 6.9. Therefore the final priority order remains unchanged. However, assuming a change of weights associated with the DDoS attack performed on the Utility Partner, and to maintain the initial values for the others, the final result drastically changes. Assuming the following choice: 0.7 for the Asset Value, 0.2 for the Exploitation Severity, and 0.1 for the Attack Strength. In this case, the Impact value becomes 5.3, which is lower than the initial Impact for the attack to the HMI, which was 5.6. In this case, the priority order proposed starts with the deployment of a new instance to filter the communications to the HMI and then the update of the policies for the packet filter protecting the Utility Partner.

Proceeding with the calculation of the risk level, the last two parameters to take into consideration are Attack Frequency and Defence Efficiency. For the first parameter, it is important to note that obtaining precise, organization-specific data would require a complete and reliable internal database recording all past incidents, their nature, and recurrence. Since such detailed information is rarely available, especially in the context of this validation, an external data source has been used instead. In particular, the Verizon Data Breach Investigation 2025 Report has been used as a reference. According to its findings, approximately 20% of recorded incidents originate from the exploitation of vulnerabilities, which, out of a total

of 9.891 incidents, corresponds to roughly 1.978 events. These values are reported across 10,000 companies, which means the mean number of such events per company is 0.198. This same value is applied uniformly to all the attack vectors considered in the validation, since each of them exploits a specific vulnerability within the system. Due to the formulation of the Poisson probability used in the model, there is no need to convert this value to the 0–10 scale applied to other parameters, as it can be directly employed in its original form for the computation of the likelihood.

Finally the Defence Efficiency parameter needs to be computed. Similarly to the Attack Strength values, the efficiency of the defence in place in the system requires the estimation on real and live data obtained from the physical system. However, the absence of a physical system under investigation makes it necessary to formulate assumptions about the defence mechanisms currently protecting the network. For the attack vector targeting the Utility Partner, it can be reasonably assumed that the firewall instance positioned between the attacker and the victim node is already capable of detecting and filtering approximately 40% of the malicious traffic flows directed to the service node. Consequently, the value assigned to the Defence Efficiency on the scale 0–1 is set to 0.4. Conversely, for the Public Database scenario it is supposed that the packet filter protecting it is unable to recognise the malicious traffic and allows it to pass without any filtering. Thus, in this case, the Defence Efficiency parameter is set to 0. Finally, for the attack scenario against the HMI interface no dedicated defence mechanism is considered to be in place, and for this reason the Defence Efficiency value is again set to 0, reflecting the total absence of protective measures for that component.

The computation of the Likelihood value for each attack vector is $1-e^{-\lambda}$, where λ is obtained from the product between the Attack Frequency, in this case 0.198 for all of them, and the complement of the controls' efficacy, respectively 0.6, 1 and 1. Given the parameters above the Likelihood for the three scenarios are:

- for the first attack scenario the value is $1-e^{-0.198*0.6} = 1-e^{-0.12} = 1-0.89 = 0.11$
- for the attack vector targeting the Public Database the value is equal to $1 - e^{-0.198*1} = 1 - e^{-0.198} = 1 - 0.82 = 0.18$
- for the attack scenario targeting the HMI interface the value is the same as above, $1 - e^{-0.198*1} = 1 - e^{-0.198} = 1 - 0.82 = 0.18$

To maintain the coherence with the Impact values, all the resulting numbers must be converted to the scale from 0 to 10. Therefore, the final Likelihood values are 1.1, 1.8, and 1.8.

Finally, the risk level for each attack scenario is computed as the product of the Impact and the Likelihood. As an example of implementation assume to consider the case in which the weights of the factors linked to the impact values are all equal to 0.33. A summary of the values of the parameters involved is present in Table 6.2. As reported in the table, the resulting risk levels obtained for the selected attack scenarios are 7.6, 7.7, and 10.1. Given these numbers, the prioritization list for the FATO framework can be established based on the highest risk scores. Accordingly,

the first action should be aimed at protecting the HMI interface, which presents the highest combined risk. The second priority involves updating the security policies in the firewall instance protecting the Public Database. Finally, the last action should address the update of the security policies in the packet filter instance protecting the Utility Partner. It is worth noting that, although the Impact values alone would have suggested a different order, the inclusion of the Likelihood factor modifies the priorities. This ensures that the HMI interface, being the most critical node, receives immediate attention, followed by other components according to their assessed risk.

Summary			
Victim	Impact Value	Likelihood Value	Risk Level
Utility Partner	6.9	1.1	7.6
Public Database	4.3	1.8	7.7
HMI	5.6	1.8	10.1

Table 6.2. Risk Parameters Summary

This example clearly highlights how relying solely on the impact score could have overlooked the temporal danger posed by attacks, as it underlines that an attack with a moderate impact and higher likelihood can be more critical than a high-impact, low-likelihood event. By integrating both Impact and Likelihood, the risk assessment captures a more realistic picture of the threat landscape. Therefore, the computed risk levels serve as a crucial guide for the implementation of protective measures, as the derived prioritization prevents the misallocation of efforts.

Chapter 7

Conclusions and Future Work

In this thesis work, a novel approach for optimizing the automatic reconfiguration of Network Security Functions (NSFs) has been studied and implemented. The method proposes to exploit preventive calculations of risk assessment models to define a precise and verifiable order of priority with which to reconfigure the network in the event of complex, multi-vector, or multi-stage attacks. In particular, the approach is designed to establish a priority order for the numerous configuration steps that occur in the intermediate stages of a transient. This emphasis is crucial because the execution of multiple reconfiguration steps without an optimal sequence may represent a potential window of vulnerability. For this reason, the framework chosen for validation is the FATO (FirewAll Transients Optimizer) framework, whose primary goal is to optimise the order of firewall reconfigurations while maximising the number of secure intermediate states. By integrating the proposed risk-driven prioritisation into FATO, the system gains the ability to react more intelligently to evolving attack scenarios. The work has been contextualised within a critical infrastructure environment, where the responsiveness and optimality of the solution are fundamental elements to ensure continuity, safety, and resilience of operations.

An in-depth research on the current landscape of network security automation has been carried out to understand both the strengths and the critical issues of the frameworks already available. In particular, the analysis of the FATO framework confirmed the optimality of its scheduling mechanism, which effectively minimises the number of insecure intermediate stages during firewall reconfigurations. However, the assessment also revealed that the framework, in its original form, lacks the capability to handle scenarios involving multi-stage and multi-vector attacks. This limitation highlights the necessity of developing an extension of the model capable of integrating a prioritization order based on the risk associated with each attack scenario, thus enabling FATO to respond effectively to more complex and evolving attack patterns.

To address this problem, an analysis of the most well-known risk assessment models has been carried out. The investigation of risk assessment models built to prevent and classify the risks linked to cyberattacks has made it possible to study a model that allows the reconfiguration of the NSFs to be organized in order to first solve the most serious problems in any scenario.

The result is a semi-quantitative model where aspects from the NIST risk assessment and the Open FAIR risk analysis are merged and customized in order to be effective for reacting to real-time attacks. The approach aims to compute the risk level associated with each attack, or step in case of multi-stage attacks, by analysing both its impact and likelihood of recurrence. An extensive description of the parameters used for the formulation of the two factors has been provided. For each of them, examples were given of both internal sources and external sources from which to extrapolate the information and data necessary for calculating the interesting values. For the computation of the *Impact* that each attack vector may have on the organization, a weighted sum has been proposed. The weights associated with the three parameters, Asset Value, Exploitation Severity, and Attack Strength, are left up to the user to choose, with some restrictions, based on the organization's internal priorities. For the *Likelihood* parameter, instead, a Poisson distribution has been chosen to estimate the probability that the same attack scenario will recur in the future, given the security measures in place.

Finally, the implemented approach has been validated within a critical infrastructure environment, simulating vulnerabilities and attack sources drawn from real components of an operational SCADA system. This test-bed reproduced realistic conditions under which cyberattacks may occur, allowing the model to be assessed on its ability to compute risk levels and drive prioritisation decisions in a practical context. For each attack vector considered, the parameters required to calculate the Impact value have been derived using both internal sources, such as the CVSS score of the mentioned vulnerabilities, and external data sources, such as the IBM Cost of Data Breaches 2024 report. Several experiments have been performed by varying the weights associated with each parameter to reflect different organizational priorities and sensibilities. The same analysis has been performed to estimate the parameters tied to the evaluation of the Likelihood value for each attack scenario. In this case only external sources, such as the Verizon Data Breach Investigations 2025 report, and likely hypotheses have been included. From the resulted values the Risk Level of each attack scenario has been computed. The resulting numeric values have been provided to the model to determine the prioritisation order of the required reconfiguration steps. This comparative analysis provided insights about the model's ability to adapt to different configurations, showing its flexibility in adjusting to changing contexts while maintaining a structured and verifiable prioritisation of reconfiguration actions.

To conclude, during the development of this thesis work, an innovative approach for calculating an attack's risk in a computer network has been studied. It represents an important first step for future developments and applications in the field of prioritized security reconfiguration in critical infrastructures.

Future work may further investigate and improve this approach by extensively studying its behaviour on real physical networks and gaining a greater comprehension of its effectiveness and time consumption. In particular, a possible continuation of the research is to analyse how the introduction of real-world parameters for calculating the Likelihood parameter can influence the final result. Moreover, other activities may exist outside the restricted scope of transients and extend the same approach to the reconfiguration issue presented in other automation approaches.

Bibliography

[1] D. Bringhenti, S. Bussa, R. Sisto, and F. Valenza, "Atomizing firewall policies for anomaly analysis and resolution," *IEEE Trans. Dependable Secur. Comput.*, vol. 22, no. 3, pp. 2308–2325, 2025. [Online]. Available: https://doi.org/10.1109/TDSC.2024.3495230

[2] D. Bringhenti, G. Marchetto, R. Sisto, and F. Valenza, "Automation for network security configuration: State of the art and research trends," *ACM Comput. Surv.*, vol. 56, no. 3, pp. 57:1–57:37, 2024. [Online]. Available: https://doi.org/10.1145/3616401

[3] D. Bringhenti, G. Marchetto, R. Sisto, F. Valenza, and J. Yusupov, "Automated optimal firewall orchestration and configuration in virtualized networks," in *NOMS 2020 - IEEE/IFIP Network Operations and Management Symposium, Budapest, Hungary, April 20-24, 2020*. IEEE, 2020, pp. 1–7. [Online]. Available: https://doi.org/10.1109/NOMS47738.2020.9110402

[4] ——, "Automated firewall configuration in virtual networks," *IEEE Trans. Dependable Secur. Comput.*, vol. 20, no. 2, pp. 1559–1576, 2023. [Online]. Available: https://doi.org/10.1109/TDSC.2022.3160293

[5] D. Bringhenti and F. Valenza, "Greenshield: Optimizing firewall configuration for sustainable networks," *IEEE Trans. Netw. Serv. Manag.*, vol. 21, no. 6, pp. 6909–6923, 2024. [Online]. Available: https://doi.org/10.1109/TNSM.2024.3452150

[6] D. Bringhenti, R. Sisto, and F. Valenza, "Automating VPN configuration in computer networks," *IEEE Trans. Dependable Secur. Comput.*, vol. 22, no. 1, pp. 561–578, 2025. [Online]. Available: https://doi.org/10.1109/TDSC.2024.3409073

[7] F. Pizzato, D. Bringhenti, R. Sisto, and F. Valenza, "Automatic and optimized firewall reconfiguration," in *NOMS 2024 IEEE Network Operations and Management Symposium, Seoul, Republic of Korea, May 6-10, 2024*. IEEE, 2024, pp. 1–9. [Online]. Available: https://doi.org/10.1109/NOMS59830.2024.10575212

[8] D. Bringhenti and F. Valenza, "Optimizing distributed firewall reconfiguration transients," *Comput. Networks*, vol. 215, p. 109183, 2022. [Online]. Available: https://doi.org/10.1016/j.comnet.2022.109183

[9] National Institute of Standards and Technology, "Guide for conducting risk assessments," U.S. Department of Commerce, Gaithersburg, MD, NIST Special Publication 800-30 Revision 1, September 2012. [Online]. Available: https://doi.org/10.6028/NIST.SP.800-30r1

[10] ——, "Cybersecurity framework manufacturing profile," U.S. Department of Commerce, Gaithersburg, MD, NIST Interagency/Internal Report (NISTIR)

8183, May 2017. [Online]. Available: https://doi.org/10.6028/NIST.IR.8183

[11] ——, "Minimum security requirements for federal information and information systems," U.S. Department of Commerce, Gaithersburg, MD, FIPS Publication 200, March 2006. [Online]. Available: https://csrc.nist.gov/publications/detail/fips/200/final

[12] W. Stallings, *Effective Cybersecurity: A Guide to Using Best Practices and Standards*. Boston, MA: Pearson Education, 2019, pp. 74–135.

[13] National Institute of Standards and Technology, "Standards for security categorization of federal information and information systems," U.S. Department of Commerce, Gaithersburg, MD, FIPS Publication 199, February 2004. [Online]. Available: https://doi.org/10.6028/NIST.FIPS.199

[14] The Open Group, *The Open Group Standard: Risk Analysis (O-RA), Version 2.0.1*. The Open Group, November 2021. [Online]. Available: https://publications.opengroup.org/standards/open-fair-standards/c250

[15] ——, *The Open Group Standard: Risk Taxonomy (O-RT), Version 3.1*. The Open Group, May 2025. [Online]. Available: https://publications.opengroup.org/standards/open-fair-standards/c251

[16] J. Wang, M. Neil, and N. E. Fenton, "A bayesian network approach for cybersecurity risk assessment implementing and extending the FAIR model," *Comput. Secur.*, vol. 89, 2020. [Online]. Available: https://doi.org/10.1016/j.cose.2019.101659

[17] A. Le, Y. Chen, K. K. Chai, A. Vasenev, and L. Montoya, "Incorporating FAIR into bayesian network for numerical assessment of loss event frequencies of smart grid cyber threats," *Mob. Networks Appl.*, vol. 24, no. 5, pp. 1713–1721, 2019. [Online]. Available: https://doi.org/10.1007/s11036-018-1047-6

[18] K. Foerster, S. Schmid, and S. Vissicchio, "Survey of consistent software-defined network updates," *IEEE Commun. Surv. Tutorials*, vol. 21, no. 2, pp. 1435–1461, 2019. [Online]. Available: https://doi.org/10.1109/COMST.2018.2876749

[19] S. Bussa, R. Sisto, and F. Valenza, "Security automation using traffic flow modeling," in *2022 IEEE 8th International Conference on Network Softwarization (NetSoft)*, 2022, pp. 486–491. [Online]. Available: https://doi.org/10.1109/NetSoft54395.2022.9844025

[20] D. Bringhenti, S. Bussa, R. Sisto, and F. Valenza, "A two-fold traffic flow model for network security management," *IEEE Trans. Netw. Serv. Manag.*, vol. 21, no. 4, pp. 3740–3758, 2024. [Online]. Available: https://doi.org/10.1109/TNSM.2024.3407159

[21] H. Yang and S. S. Lam, "Real-time verification of network properties using atomic predicates," *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 887–900, 2016. [Online]. Available: https://doi.org/10.1109/TNET.2015.2398197

[22] D. Bringhenti, F. Pizzato, R. Sisto, and F. Valenza, "A looping process for cyberattack mitigation," in *2024 IEEE International Conference on Cyber Security and Resilience (CSR)*, 2024, pp. 276–281. [Online]. Available: https://doi.org/10.1109/CSR61664.2024.10679501

[23] ——, "Autonomous attack mitigation through firewall reconfiguration," *International Journal of Network Management*, vol. 35, no. 1, p. e2307, 2025, e2307 nem.2307. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/nem.2307

[24] A. Giacchero, F. Giordano, and M. Schiraldi, "From business continuity to design of critical infrastructures: ensuring the proper resilience level to datacentres," *International Journal of Engineering & Technology*, vol. 5, no. 4, pp. 3544–3553, 2013. [Online]. Available: https://hdl.handle.net/2108/100984

[25] P. Mcilwee, K. Penuel, M. Statler, and R. Hagen, "Business continuity management," *Encyclopedia of crisis management*, pp. 74–76, 2013.

[26] S. Torabi, H. Rezaei Soufi, and N. Sahebjamnia, "A new framework for business impact analysis in business continuity management (with a case study)," *Safety Science*, vol. 68, pp. 309–323, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925753514001027

[27] M. Aghabegloo, K. Rezaie, S. A. Torabi, and M. Yazdani, "Integrating business impact analysis and risk assessment for physical asset criticality analysis: A framework for sustainable operations in process industries," *Expert Systems with Applications*, vol. 241, p. 122737, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417423032396

[28] C. Păunescu, M. C. Popescu, and L. Blid, "Business impact analysis for business continuity: Evidence from romanian enterprises on critical functions," *Management & Marketing*, vol. 13, no. 3, pp. 1035–1050, 2018.

[29] H. Hassel and A. Cedergren, "Integrating risk assessment and business impact assessment in the public crisis management sector," *International Journal of Disaster Risk Reduction*, vol. 56, p. 102136, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212420921001023

[30] *Common Vulnerability Scoring System v3.1: Specification Document*, FIRST.Org, Inc., 2019, available online: https://www.first.org/cvss/v3.1/specification-document (accessed on 2 September 2025).

[31] D. Waltermire and K. Scarfone, "Nist special publication 800-126 revision 2: The technical specification for the security content automation protocol (scap)," National Institute of Standards and Technology, NIST Special Publication 800-126r2, 2011. [Online]. Available: https://doi.org/10.6028/NIST.SP.800-126r2

[32] H. Hu, "Poisson distribution and application," *A Course in Department of Physics and Astronomy; University of Tennessee at Knoxville: Knoxville, TN, USA*, 2008.

[33] D. Ranathunga, M. Roughan, H. X. Nguyen, P. Kernick, and N. J. G. Falkner, "Case studies of SCADA firewall configurations and the implications for best practices," *IEEE Trans. Netw. Serv. Manag.*, vol. 13, no. 4, pp. 871–884, 2016. [Online]. Available: https://doi.org/10.1109/TNSM.2016.2597245

[34] G. Yadav and K. Paul, "Architecture and security of SCADA systems: A review," *Int. J. Crit. Infrastructure Prot.*, vol. 34, p. 100433, 2021. [Online]. Available: https://doi.org/10.1016/j.ijcip.2021.100433

[35] N. Gaudet, A. Sahu, A. E. Goulart, E. Rogers, and K. Davis, "Firewall configuration and path analysis for smartgrid networks," in *2020 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR)*, 2020, pp. 1–6.

[36] *Information Disclosure, Information Tampering and Denial of Service (DoS) Vulnerability in GENESIS64 and MC Works64*, FIRST.Org, Inc., 2024, available online: https://www.cve.org/CVERecord?id=CVE-2024-7587 (accessed on 11 September 2025).

[37] *SQL Injection in langchain-ai/langchain*, FIRST.Org, Inc., 2024, available online: https://www.cve.org/CVERecord?id=CVE-2024-8309 (accessed on 11 September 2025).

[38] *iniNet Solutions SpiderControl SCADA PC HMI Editor Path Traversal*, FIRST.Org, Inc., 2024, available online: https://www.cve.org/CVERecord?id=CVE-2024-10313 (accessed on 11 September 2025).

[39] *Endpoint Denial Of Service*, The MITRE Corporation, 2025, available online: https://attack.mitre.org/techniques/T1499/ (accessed on 11 September 2025).

[40] *Exploitation for Privilege Escalation*, The MITRE Corporation, 2025, available online: https://attack.mitre.org/techniques/T1068/ (accessed on 11 September 2025).