SAPIENZA
UNIVERSITÀ DI ROMA

INGEGNERIA DELL'INFORMAZIONE, INFORMATICA E STATISTICA

Security Governance

Author:
Alessio Amatucci

Professor:
Silvia Bonomi

Academic year 2021/2022

Contents

Disclaimer

This document was created to provide an overview in a single collection of the Security Governance program. Note that this document may contain errors: whether they are typing, grammatical or content. In fact, I want to clarify that is a **personal** review of the slides and the contents explained by professor Silvia Bonomi during the lessons of the Security Governance course in A.Y. 2021/2022, which may differ from the real intended meaning by the teacher.

1 Introduction

Governance is all of the processes of governing through the laws, norms, power or language of an organized society.

1.1 Enterprise Governance

Enterprise governance is the structure and relationships that control, direct, or regulate the performance of an enterprise (and its) projects, portfolios, infrastructure, and processes. Enterprise Governance is composed by:

- **Corporate Governance**: Corporate governance is the system of rules, practices and processes by which a firm is directed and controlled.
- **Business Governance**: Business Governance is a set of policies and business processes that set the way that the organisation's business is run.

1.1.1 Corporate Governance

Analyzing the concept of Corporate Governance we can define the action of "**directing**" the act of establishing responsibilities and planning, while to "**control**" the act of ensuring implementation, control of results and application to compliance. Employees of an organization can be viewed within a pyramid and divided into 3 levels as shown in Figure 1:

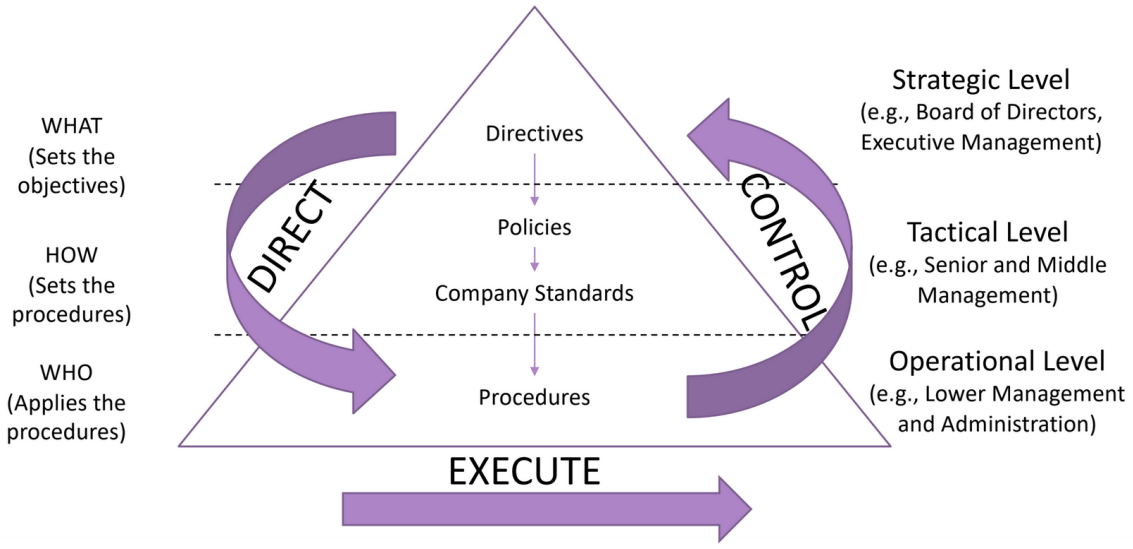


Figure 1: Corporate Governance Direct/Control loop.

On the top we have the **Executive Management** that make up the **Strategic Level** and answer the question "What" needs to be done to set an objective. Then at the second level we have the **Senior and Middle Management** which constitute the **Tactical Level** and answer the question "How" to set up the procedures. Finally, at the third level we have the **Operational Level** which is composed of **Lower Management and Administration** and the question is answered "Who" applies the procedures.

The IT Governance Institute redefine the concept of Enterprise (and thus Corporate) Governance as follow:
"*Enterprise governance is a set of responsibilities and practices exercised by the board and executive management with the goal of providing strategic direction, ensuring that objectives are achieved, ascertaining that risks are managed appropriately and verifying that the enterprise's resources are used responsibly.*" [1]

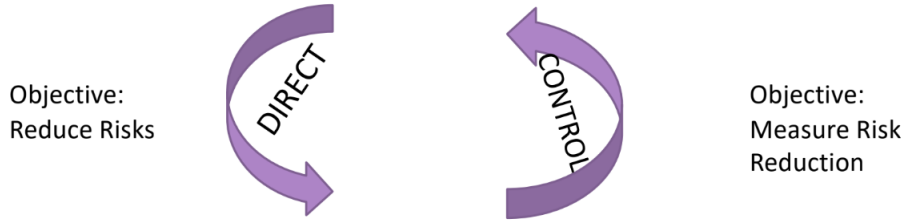


Figure 2: Direct/Control loop.

The cycle represented in Figure 2 shows how the directives go from top to bottom in which they are executed, where the control action is then performed and it is verified that the objective has been achieved. Obviously the goal is also to try to stabilize the process in order to stop the alteration.

1.2 Enterprise Governance Decomposition

Corporate Governance and Business Governance have Governance Sub-system, some of these are Financial Governance, HR Governance, IT Governance and so on, we will focus only on the latter, in fact for Gartner[1] "*IT Governance (ITG) is defined as the processes that ensure the effective and efficient use of IT in enabling an organization to achieve its goals.*"

1.3 IT Governance

IT based systems causes serious risks to a company as it manages and stores many electronic assets like data and information stored, data and information transmitted over the network and all the systems and application required to store, transmit and process data and information. So it's clear that electronic assets are exposed to many threats aiming at compromising their CIA properties (Confidentiality, Integrity and Availability).

1.4 Security Governance

According to UK National Cyber Security Centre[2] "*Security governance is the means by which you control and direct your organisation's approach to security*". When done well, security governance will effectively coordinate the security activities of your organisation. It enables the flow of security information and decisions around your organisation, but to do so it is necessary to have an overall view of the entire enterprise.
To answer the question: Which approach to security governance is right for me? We can say that there is no "one size fits all" approach to security governance. It is understandable, however, that a large company must have a formalized security framework, with clearly defined roles and business processes, on the contrary in the case where the company is a smaller reality it is possible to use an informal approach to directing, controlling and making security decisions, in both cases we face the common problem of the availability of limited economic resources to have high security, therefore it is necessary to develop a model in which all the critical points of infrastructure are defined and to regulate the security level. Certainly, further security measures come into play when it is necessary to consider the exchange of sensitive information with external parties.
We should care about Security Governance because every year there is an increment in the number of vulnerabilities that can be exploited and it is important to define the difference between **breach** and **incident**:

- A breach tell us how break confidentiality.
- An incident tells us how the attack ends and breaks integrity and availability.

[1] https://en.wikipedia.org/wiki/Gartner
[2] https://en.wikipedia.org/wiki/National_Cyber_Security_Centre_(United_Kingdom)

2 A model for Information Security Governance

In NIST SP 800-100, where SP stay for Special Publication, Information Security Governance is defined as:
"*The process of establishing and maintaining a framework and supporting management structure and processes to provide assurance that information security strategies are aligned with and support business objectives, are consistent with applicable laws and regulations through adherence to policies and internal controls, and provide assignment of responsibility, all in an effort to manage risk.*"
ISG must ensure cost-effectiveness:

- There must be a balance between the cost of protecting electronic resources and the risk to which these resources are exposed.
- No overprotection causing unnecessary expenses.
- No underprotection causing risk to materialize and impact the company.

A good (IT) Risk management strategy is therefore mandatory to implement a good ISG strategy. The model introduced by Von Solms [2] shown in Figure 3 provides a Front Dimension (Core Part) and a Depth Dimension (Expanded Part).

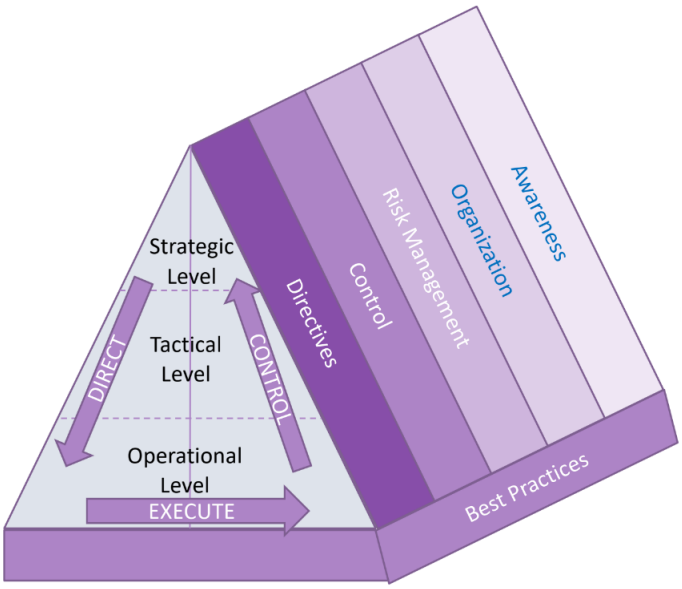


Figure 3: ISG Model introduced by Von Solms R. in 2006 from [2].

The arrow that goes from top to bottom tells us that from the highest layer the things to do are decided in broad terms and are gradually made more detailed and appropriately documented as we arrive at the layer in which they are put into practice (Direct arrow). On the contrary, the arrow that proceeds towards the top indicates that the reports produced by the operational level are very detailed and rich, while the organization's board only receives stringent considerations to see if the policies imposed by them are working properly or not and possibly make other decisions (Control arrow).

2.1 The Front Dimension (Core Part)

It represents the execution of processes and actions and the influence of the Direct and Control loop on these processes. It is based on two core principles:

1. It covers the 3 well known level of management: Strategic, Tactical and Operational, which are made up of people who have different responsibilities and skills.
2. Across these 3 levels, there are very distinct actions where the granularity of information is highest at the lowest level (Operational Level).

2.1.1 The Direct Part

1. The **Strategic layer** identify assets, their relevance and their required level of protection.
   INPUT: External factors (legal and regulatory prescriptions and other external risks) and/or Internal factors (company's strategic vision, IT role, competitiveness, etc).
   OUTPUT: a set of Directives indicating (at high level) what the Board expects must be done as far as the protection of the company's information assets is concerned.

2. In the **Tactical layer** directives are "expanded" into sets of relevant information security policies, company standards and procedures.
   INPUT: The directives of the Strategic layer.
   OUTPUT: Policies, procedures and standards.

3. In the **Operational layer** inputs are expanded into sets of administrative guidelines and administrative procedures and technical measures are physically implemented and managed.
   INPUT: The policies, standards and procedures by the Tactical layer.
   OUTPUT: Operating procedures specifying how things must be done. It forms the basis of execution on the lowest level.

2.1.2 The Control Part

To properly Control (manage) we need to measure and for this we need to know which information and data to collect. This "measurability" characteristic must be at the centre of all directives, policies, standards and procedures produced during the "Direct" part of the model.

1. In the **Operational layer** measurement data is extracted from a wide range of entities (either automatically or manually).
   INPUT: Measurements related to data extracted from a series of entities such as sensors, IDS, IPS etc.
   OUTPUT: Specialized reports can be created on this level using this extracted operational data.

2. In the **Tactical layer** measurement and monitoring against the requirements of the relevant policies, procedures and standards.
   INPUT: With the operational layer report, it is necessary to decide whether the policies, standards and procedures are suitable for protecting assets.
   OUTPUT: Tactical Management reports, indicating levels of compliance and conformance.

3. In the **Strategical layer** the Control will be a situational Awareness.
   INPUT: Tactical management reports and understand how things are going within the organization.
   OUTPUT: Reports reflecting compliance and conformance to relevant directives including risk considerations.

2.2 Best Practices

Best Practices (or Standards or Guidelines) are a set of documents reporting experiences and solutions experienced by experts in the field of Information Security and provide an internationally accepted framework that can be used as building block for ISG.

2.2.1 ISO 27002

It is an International Standard, a "guideline" document, and advises companies on what they should have in place as far as their Information Security Management is concerned, in order to follow "Best Practice". It guides a company to structure its Information Security Management according to the experience of other companies. This document contains 14 security control clauses, collectively containing a total of 35 main security categories and 114 controls. Each clause defining security controls contains one or more main security categories where each of them contains:

- a control objective stating what needs to be achieved;
- one or more controls that can be applied to achieve the control objective, where each one has:
  - **Control**: the specific control statement, to satisfy the control objective.

- **Implementation Guidance**: Provides more detailed information to support the implementation of the control and meeting the control objective.
- **Other Information**: Provides further information that may need to be considered, like references to other standards.

So we understand that ISO 27002 allows us to answer the question HOW we should do the policies, but we do not define them in technical language.

2.2.2 ISO 27001

It is an International Standard and specifies the requirements for establishing, implementing, maintaining and continually improving an Information Security Management System (ISMS) within the context of the organization. It also includes requirements for the assessment and treatment of information security risks tailored to the needs of the organization. It is generic and intended to be applicable to all organizations, regardless of type, size or nature, but unlike ISO 27002, this is a standard for obtaining a certification and therefore is much more rigorous and specific, it indicates in detail what an organization must have or do to be compliant.

2.3 Directives

Security Policy documents are required from Standards and Best Practices, from ISO 27002 we have *5.1 Management direction for information security* which the objective is to provide management direction and support for information security in accordance with business requirements and relevant laws and regulations. This set of policies should be defined, approved by management, published and communicated to employees and relevant external parties.
In order to comply with the requirements of having a documented Direct process, it is important to define a methodology to create, manage and distribute policy related documents and this aim is achieved with the Information Security Policy Architecture (ISPA) and related Documents.

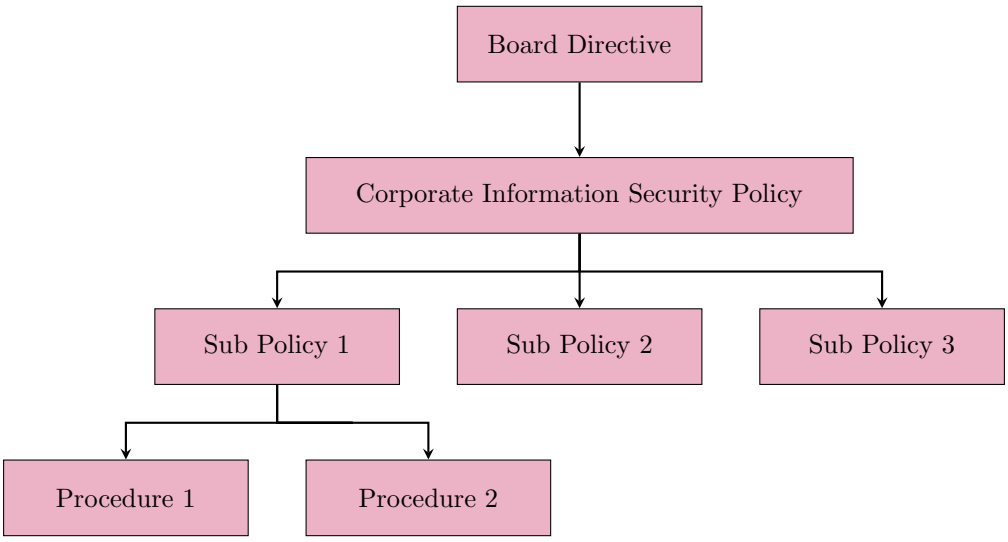


Figure 4: Structure of Information Security Policy Architecture.

As you can see in Figure 4 it is a tree where each level refers to a class of documents. The **Board Directive** must contain the set of IT assets identified by the Board and the mandate from the Board to protect these assets. Then we have the Corporate Information Security Policy (CISP) which is a high-level document providing a basis for all lower-level documents related to Information Security, supported by a set of Sub-Policies that may differ from company to company and defines specific important aspects in more detail. Finally, we have the procedures that specify how a sub-policy must be implemented.

2.3.1 The Corporate Information Security Policy (CISP)

Guidelines to create a proper CISP are:

1. The CISP must indicate Board and executive management support and commitment and it must be clear that the CISP flows from a higher-level directive.
2. The CISP must be accepted and signed by the CEO or equivalent officer.
3. The CISP must not be a long document, nor must it be written in a technical form. The maximum length should be about four to five pages, and it must contain high-level statements concerning Information Security.
4. The CISP should not change very often, and must be 'stable' as far as technical developments and changes are concerned.
5. For the reason mentioned above, the CISP must not contain any references to specific technologies, and must be "technology neutral".
6. The CISP must indicate who is the owner of the Policy and what the responsibilities of other relevant people are.
7. The CISP must clearly indicate the Scope of the Policy, that is, all people who will be subject to the Policy.
8. The CISP must refer to possible (disciplinary) actions for non-conformance to the it and its lower-level constituent policies.
9. The CISP must be distributed as widely as possible in the company and must be covered in all relevant awareness courses.
10. The CISP must have a Compliance Clause.

2.3.2 Sub-policies

The set of Sub-Policies may differ from company to company, but usually the following ones will be defined for everybody:

- A Malicious Software Control Policy (Antivirus Policy).
- An Acceptable Internet Usage Policy.
- An Acceptable Email Usage Policy.
- A Logical Access Control Policy.
- A Disaster Recovery (Backup) Policy.
- A Remote Access Control Policy.
- A Third Party Access Control Policy.

2.4 Control

Compliance check is the main scope of the Control part of the model and in ISO 27002 it is defined with paragraph 18 "Compliance". Managers should regularly review the compliance of information processing and procedures within their area of responsibility with the appropriate security policies, standards and any other security requirements. The organization's approach to managing information security and its implementation should be reviewed independently at planned intervals or when significant changes occur. This review should be carried out by parties independent of the area under consideration; here the timing is essential, because if the control is too fast compared to the management phase, then it is likely that the policies do not have time to become effective, on the contrary, if the control is too late, monitoring could involve an environment radically different from that for which the policies were intended and these could have changed. Unfortunately, specifying compliance clauses is not an exact science and very few guidelines are available like:

- Compliance Clauses must be clear and precise.
- Compliance Clauses should express a way to measure its satisfaction.

2.5 Risk Management

Risk Management is the process to identify and assess all potential risks as well as introducing controls that should mitigate all these risks to acceptable low levels. Today, in most circumstances, risk has two factors associated with it:

- a probability or frequency;
- a magnitude of gains or losses (impact).

Thus, the aim of risk management is to determine what the impact will be if the risk does materialize and how often (probability or frequency) this risk might materialize.
The risk can be quantified with the following formula:

$$Risk = Likelihood \cdot impact = (Thread \cdot Probability) \cdot Impact$$

You can reduce the risk by reducing the potential impact, frequency of the risk or a combination of both. All risks that could possibly have a negative effect on the well-being of the organization are definitely the responsibility of management. Thus, all levels of organizational management should be involved in the process of Risk Management, so it is imperative that information and IT-related risks are managed in an integrated manner with business risks.

2.6 Organization

In any company, the way Information Security is organized is very important. A good approach is to have at least two distinct components in the organization:

1. One looking at day-to-day operational aspects.
2. One responsible for the compliance monitoring function.

The IT Risk Management Committee is a sub-committee of the Audit Committee, and reports through it directly to the Board. The following diagram, Figure 5, indicates how the full Information Security function can be organized in a company:

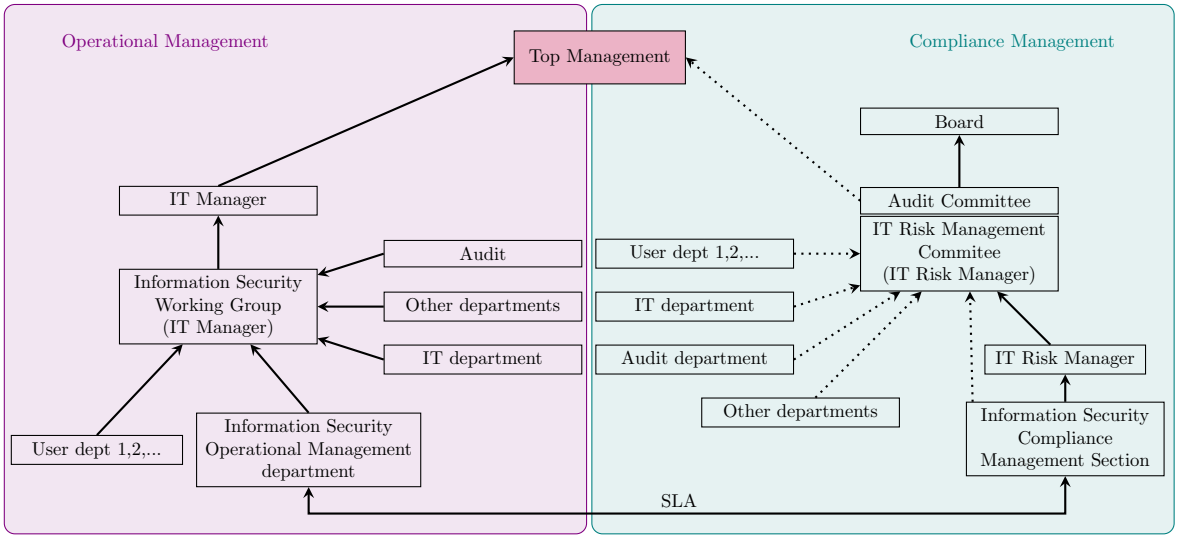


Figure 5: Organizing Information Security Governance from [3].

There is a need for a clear separation between those who work for the direct part and those who work for the control part, for this reason we have two types of Management:

- **Information Security Operational Management** composed of all the resources who follow the **operational** aspects day after day.
- **Information Security Compliance Management** composed of the resources that follow the matters relating to **compliance** and monitoring.

2.7 Awareness

Since IT security procedures must be drafted and published in such a way that they are transmitted to all users of the organization, it is essential that workers are aware of and trained on policies and procedures. All of this is formalized in a program known as Information Security Education Training and Awareness (SETA) whose goal is to provide users with training on the safe way to work and raise their level of awareness of the importance of protecting IT.
SETA Program is based on 3 terms:

1. **Education**: It is based on **why** we solve the problem.
2. **Training**: It is based on **how** the problem is solved.
3. **Awareness**: It is based on **what** the problem is.

During the completion of the SETA program, it makes sense to talk about the Conscious Competence Learning Model (Figure 6) with which we analyze the stages through which the individual employee passes.

1. **Unconscious Incompetence** where the subject is a person does not realize that he/she does not know how to do certain thing and it is important to make the individual aware that he/she is incompetent as far as that specific task is concerned.
2. **Conscious Incompetence** it is important that the employee is trained to do his/her job in a secure manner.
3. **Conscious Competence** the employee knows what to do and how to do it to ensure his/her job is done in a secure manner, but still needs to concentrate in order to perform the necessary procedures correctly.
4. **Unconscious Competence** the employee through sufficient practice and experience makes all practices and skills related to information security inherent in his normal daily actions and behaviors.

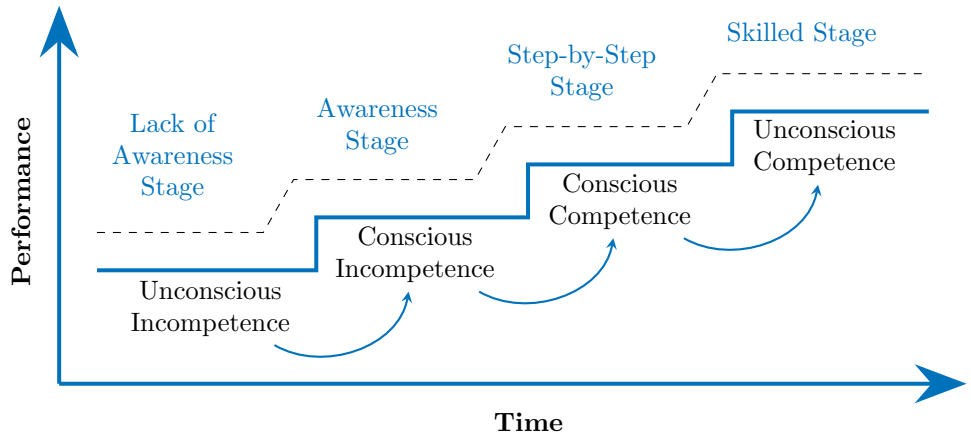


Figure 6: Conscious Competence Learning Model.

2.8 A Methodology for Establishing an Information Security Governance Environment

The methodology will consist of 14 steps: 9 setup steps and 5 steps in continuous cycle. It is a complete methodology as it assumes that no ISG environment exists at all and that everything must be started from scratch.

i) **Setup Phase**:

1. **Get the Board's buy-in about IT Risk Management and Information Protection**: This is a very important starting point and without it, the project will surely have problems in progressing properly. Initially, it may not be possible to get a "clean" Board Directive but that is not essential as long as there is a commitment from the Board to support the project, that is enough to start.
2. **Select some guiding Best Practices**: It is fundamental to choose some existing international Best Practices to use as a foundation for the project, and to provide motivation and direction, many of them exist and the essential aspect is to have one.
3. **Perform a basic Risk Analysis and determine all controls needed**: This is mandatory to identify actual critical assets and avoid wasting money and investment. Risk Analysis requires getting a wide spectrum of people involved as board members, line management, users, system owners etc. Ensure that the relevant controls are installed and operational.
4. **Create a Corporate Information Security Policy (CISP) and get it signed by the CEO**: A CISP must be drafted, be circulated amongst the stakeholders and then submitted to top management and now must be signed and made official company policy. This is the basis and motivation for all future steps.
5. **Create the rest of the Information Security Policy Architecture (ISPA)**.
6. **Create the organizational structure for ISG**: Specific attention must be given to the Operational Management and Compliance Management sides. This step is closely related to Step 7 and this is why it is preferable to do them in parallel.
7. **Create an initial set of Compliance/Control measures and start using these measures to create reports on all three management levels**: In this step, it is important to get buy-in from the company's Internal and External IT Auditor departments as well as the Legal department. If these are involved in Step 3 above, this process will be much easier, however, it is important to start with an initial set of measures which will form the basis of the "Control" part of the model.
8. **Create an Awareness Programme including aspects like information security job responsibilities**: This step is core to the success of the whole effort and must be performed continuously and all the documents in the ISPA should form part of the Awareness Programme.
9. **Get the cycle going - kick start the process**: At this point, the whole programme must be initiated, if one already exists, the revised one must be integrated with the existing one to get the new one operational.

ii) **Continuous Phase**:

10. **Redo the Risk Analysis from time to time to identify the possible changes in risks and controls**: Risks are dynamic, so it is important to re-do any risk analysis from time to time to ensure that the risk situation is up to date and relevant controls are in place and operational.
11. **Keep the ISPA up to date and in line with newly identified risks**: Ensure that all changed risks are reflected in the ISPA by changing the content of the ISPA and Compliance Clauses, if necessary.
12. **Refine and expand the Compliance Control measures to cater for newly identified risks, enforce compliance and keep reporting to top management**.
13. **Continue to make all users more Information Security Aware**: This process can never stop and must be enforced continuously.
14. **Return to Step 10**.

3 Cybersecurity Frameworks and Best Practices

3.1 NIST Cyber Security Framework (CSF)

NIST must identify "a prioritized, flexible, repeatable, performance-based, and cost-effective approach, including information security measures and controls that may be voluntarily adopted by owners and operators of critical infrastructure to help them identify, assess, and manage cyber risks". The Framework focuses on using business drivers to guide cybersecurity activities and considering cybersecurity risks as part of the organization's risk management processes. NIST CSF is technology neutral and was developed to improve cybersecurity risk management in critical infrastructure but it can be used by organizations in any sector or community, in fact it provides a common organizing structure assembling standards, guidelines, and practices that are working effectively today. It is also applicable to organizations relying on technology, whether their cybersecurity focus is primarily on. It provides a common taxonomy and mechanism for organizations to:

1. Describe their current cybersecurity posture.
2. Describe their target state for cybersecurity.
3. Identify and prioritize opportunities for improvement within the context of a continuous and repeatable process.
4. Assess progress toward the target state.
5. Communicate among internal and external stakeholders about cybersecurity.

3.2 Risk Management and Cyber Security Framework

To manage risk, organizations should understand the likelihood of an event occurring and the potential impacts that result from it. With this information, organizations can determine the acceptable level of risk for achieving their organizational objectives and can express this as their risk tolerance. Consists of three parts:

- Core
- Implementation Tiers
- Profile

3.2.1 NIST CSF Core

It provides a set of activities for achieving specific cybersecurity outcomes and references examples of guidelines for achieving these outcomes. The Core is not a checklist of actions to perform. It presents key cybersecurity outcomes identified by stakeholders as helpful in managing cybersecurity risk. The Core comprises four elements and can be seen as a tree in Figure 7:

1. **Functions** organize basic cybersecurity activities at their highest level:
    - **Identify**: Develop an organizational understanding to manage cybersecurity risk to systems, people, assets, data, and capabilities.
    - **Protect**: Develop and implement appropriate safeguards to ensure delivery of critical services.
    - **Detect**: Develop and implement appropriate activities to identify the occurrence of a cybersecurity event.
    - **Respond**: Develop and implement appropriate activities to take action regarding a detected cybersecurity incident.
    - **Recover**: Develop and implement appropriate activities to maintain plans for resilience and to restore any capabilities or services that were impaired due to a cybersecurity incident.
2. **Categories** are the subdivisions of a Function into groups of cybersecurity outcomes closely tied to programmatic needs and particular activities.

3. **Subcategories** further divide a Category into specific outcomes of technical and/or management activities. They provide a set of results that, while not exhaustive, help support achievement of the outcomes in each Category.

4. **Informative References** are specific sections of standards, guidelines, and practices common among critical infrastructure sectors that illustrate a method to achieve the outcomes associated with each Subcategory.

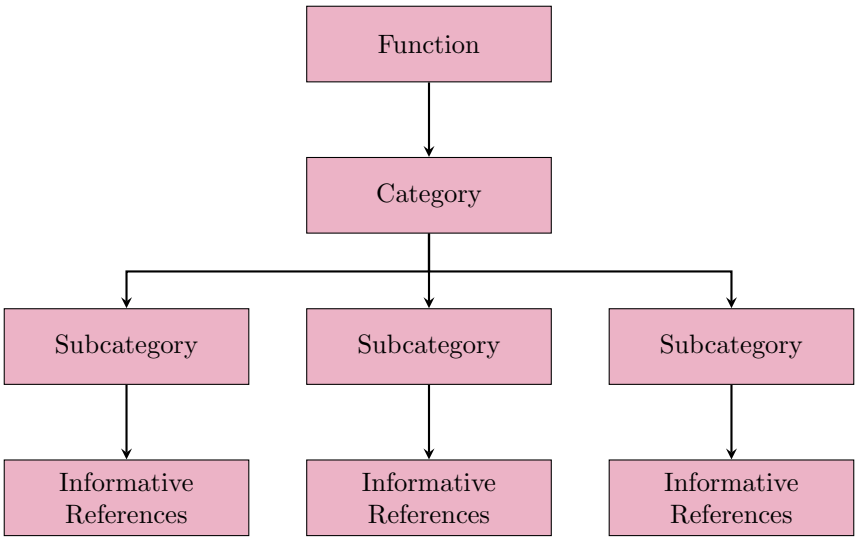


Figure 7: Overview of the NIST CSF Core's tree.

3.2.2 Implementation Tiers

The Framework implementation Tiers ("Tiers") provide context on how an organization views cybersecurity risk and the processes in place to manage that risk ranging from Partial (Tier 1) to Adaptive (Tier 4). Tiers describe an increasing degree of rigour and sophistication in cybersecurity risk management practices thus do not represent maturity levels. They are intended to support organizational decision-making on how to manage cybersecurity risk, as well as which dimensions of the organization have the highest priority and may receive additional resources. Naturally, progression to higher Tiers is encouraged when a cost-benefit analysis indicates a feasible and cost-effective reduction of cybersecurity risk. In Table 1 Tiers are reported:

Tiers	Risk Management Process	Integrated Risk Management Program	External Participation
Tier 1 Partial	• Lack of formalization • Ad hoc and sometimes reactive management • Prioritization of activities may not be directly based on by organizational risk objectives, the threat environment, or business/mission requirements	• Limited awareness of cybersecurity risk at the organizational level • Irregular, case-by-case basis due to varied experience or information gained from outside sources • The organization may not have in place information sharing processes	No Collaboration
Tier 2 Risk Informed	• Management approval but possible absence of clear policies • Prioritization of activities may not be directly based on by organizational risk objectives	• Awareness of cybersecurity risk at the organizational level but not an organization-wide approach to managing cybersecurity • Information sharing within the organization on an informal basis • Not all levels of the organization are involved • Cyber risk assessment of organizational and external assets occurs, but is not typically repeatable or reoccurring	Low Collaboration
Tier 3 Repeatable	• Formal approval and definition of policy • Regularly updated	• There is an organization-wide approach to manage cybersecurity risk	High Collaboration
Tier 4 Adaptive	• Standards and Best Practices are applied • Continuous improvement process	• There is an organization-wide approach to managing cybersecurity risk that uses risk-informed policies, processes, and procedures to address potential cybersecurity events	Full Collaboration

Table 1: Table for the Framework Implementation Tiers.

3.2.3 Profile

The Framework Profile ("Profile") is the alignment of the Functions, Categories, and Subcategories with the business requirements, risk tolerance, and resources of the organization. They can be used to describe the current state or the desired target state of specific cybersecurity activities. The Current Profile indicates the cybersecurity outcomes that are currently being achieved. The Target Profile indicates the outcomes needed to achieve the desired cybersecurity risk management goals. It is possible to carry out a gap analysis that allows you to calculate the efforts needed to move from a certain safety position to an ideal one.

3.2.4 How to Use the Framework

1. Basic Review of Cybersecurity Practices
2. Establishing or Improving a Cybersecurity Program
3. Communicating Cybersecurity Requirements with Stakeholders
4. Buying Decisions
5. Identifying Opportunities for New or Revised Informative References
6. Methodology to Protect Privacy and Civil Liberties
7. Self-Assessing Cybersecurity Risk with the Framework

3.3 National Cybersecurity Framework

The National Cybersecurity Framework is a framework developed in 2015 by Sapienza to adapt NIST-CSF to the Italian context, that is, to shape the US framework for Small and Medium Enterprises (SMEs). It is not a standard since there are no certifications, but it is a tool for self assessment, as the document contains guidelines.
The structure is completely similar to NIST-CSF but two columns are added, defined for each contextualization, i.e. the projection of the framework in the specific context:

- **Priority Level**: If you use the framework for a self assessment it is necessary to focus on the most important things for the company, it is natural that not all sub-categories have the same importance in their context.
- **Maturity levels**: Whenever something is implemented, it is possible to do it in any way. For example, in the "Assets inventory" we can record the inventory of assets in many ways, from a completely manual method to a totally automated one.

By contextualization we mean using the framework within our particular environment.

3.3.1 Mapping Data Protection Framework

The framework was changed when the directives from the EU regarding the processing of personal data (GDPR) arrived. In fact, the concepts imposed by the GDPR were taken and they wanted to put inside the framework and therefore 4 possibilities occurred:

- **Perfect placement**: there is already a correspondence between what the GDPR requires and one or more sub-categories present in the framework.
- **Partial placement**: the GDPR requires something that is already mentioned at a very high level by the framework in a sub-category or several sub-categories.
- **Not possible placement**: the GDPR requires something that is not addressed in the framework and therefore we have to implement other sub-categories.
- As regards purely legal and non-safety aspects, no matching is necessary (no sub-categories).

3.4 A comparison of CSF

The comparison takes place on standards, best practices and frameworks belonging to two different ecosystems:

- **FISMA** ecosystem: mainly used in the USA.
- **ISO 27k** ecosystem: used in the rest of the world

NIST sits between the two, highlighting only the most important aspects of one and the other.

3.4.1 Federal Information Security Modernization Act (FISMA) ecosystem

In 2002 comes the directive from the president of the USA, addressed to all federal agencies to develop a common security plan to ensure a certain level of information security. Only 10 years later this standard was completed and they moved on to phase 2, that is to the Implementation and Assessments Aids. During these 10 years a series of documents have been developed: the first two, FIPS 199 and 200, lay the foundations for all subsequent ones. In particular, the first outlines Standards for Security Categorization of Federal Information and Information Systems, while the other defines what are the Minimum Security Requirements for Federal Information and Federal Information Systems.

3.4.2 NIST Special Publications

Then we find the list of all NIST Special Publications, the NIST SP 800 series. These are highly detailed documents rich in very specific aspects or in any case restricted to particular topics. What we will focus on is **NIST SP 800-53**, which focuses on the controls that federal agencies must apply to ensure a certain level of security and privacy. It consists of a collection of checks that organizations examine for self-assessment and also provides a selection method for the checks. For ease of use in the security control selection and specification process, controls are organized into 18 families, where each family contains security controls related to the general security topic of the family. Each Security Control has the following structure:

- **Control** prescribes specific security-related activities or actions to be carried out by organizations or by information systems.
- **Supplemental guidance**: provides non-prescriptive, additional information for a specific security control.
- **Control enhancements** provides statements of security capability to:
  (i) add functionality/specificity to a control; and/or
  (ii) increase the strength of a control.
- **References** includes a list of applicable federal laws, Executive Orders, directives, policies, regulations, standards, and guidelines that are relevant to a particular security control
- **Priority and baseline allocation** provides:
  (i) the recommended priority codes used for sequencing decisions during implementation; and
  (ii) the initial allocation of security controls and control enhancements to the baselines.

Limitations: Tied to the concept of federal information systems, to the US Legislation, also it is a very complex and oriented to operational people, which is not certifiable outside the USA.

3.4.3 ISO 27k ecosystem

It is the most international framework, recognized in over 60 countries.
Limitations: Managing an ISMS is complex and requires a specific office, also it is expensive, in fact, it is rare to see SMEs adopting the standard. Typically these standards do not provide for prioritization of controls, but each of these is mandatory to receive certification. However, it remains a voluntary choice.

3.4.4 NIST SP 800-53 vs ISO 27k

The 800-53 has an appendix that provides the mapping between the controls of the two standards. All FISMA standards are very "system-oriented" while the ISOs are dedicated to the organization, so they are not very technical documents, in fact, the FISMA certification is for the "IT-System" while for the ISO it is for the "Management System". One characteristic that both of them have in common is that they are the result of years of hundreds of experts.

	NIST SP 800-53	**ISO 27001**	**CIS Control**	**NIST CSF**
Last update	Revision 5 2020	2013, reviewed and confirmed in 2019	version 8 released May 2021	version 1.1 released in April 2018
Size	Large	Medium	Medium	Small
Complexity	High	Medium	Medium	Low
International spreading	No	Yes	Yes	Medium
Target organizational level	Operational	Operational	Tactical/Operational	Strategical
Certifiability	Yes	Yes	No	No
Is it mandatory?	Yes in USA	No	No	No
Cost	-	High	Low	Low

Table 2: Comparison between standards and best practices.

4 An introduction to Risk Management

Informally a risk is the potential that something goes wrong and thereby causes harm or loss. A risk is the **likelihood** of an **incident** and its **consequence** for an **asset**.

- **Asset**: it is the core of the risk management process and it is what we want to protect. In short, an asset is something that creates value for the company or for anyone it is useful to evaluate that risk what we will call a party.
- **Likelihood**: it is the chance of something to occur.
- **Incident**: it is an event that harms or reduces the value of an asset.
- **Consequence**: it is the impact of an incident on an asset in terms of harm or reduced asset value.

4.1 Risk Management

Risk management comprises coordinated activities to direct and control an organization with regard to risk. A risk management process must be:

- **Adequate**: Proportionate to investments in safety, that is, it must neither be an exaggerated nor too small expense, but must be balanced with the company structure.
- **Efficient**: The process must be carried out efficiently or integrated to avoid wasting money and effort.
- **Effective**: At the end of the process it is possible to see that the risk has been reduced together with the possibility that something negative could happen in the future, a less structured but usable process is better than too structured, but not feasible.

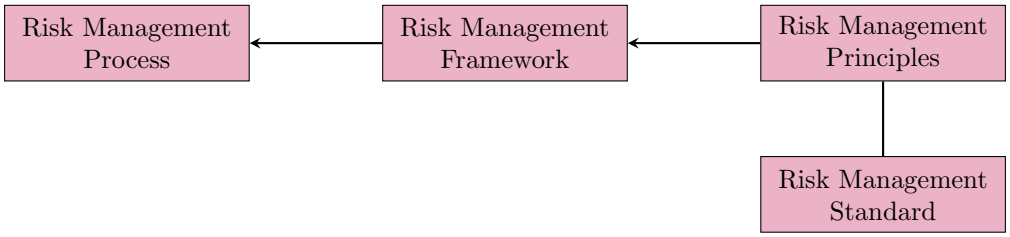


Figure 8: The Risk Management Framework, inspired by existing best practices, provides guidelines on how to implement processes, what to consider, what to focus on and supports the Risk Management Process.

4.2 ISO 31000

ISO 31000 provides guidelines on managing risk faced by organizations and can be customized to any organization and its context. It provides a common approach to managing any type of risk and is not industry or sector-specific. It can be used throughout the life of the organization and can be applied to any activity, including decision-making at all levels. ISO 31000 cannot be used for certification purposes, but it does guide internal or external audit programs.

4.2.1 Principles

The principles outlined in Figure 9 provide guidance on the characteristics of effective and efficient risk management, communicating its value and explaining its intention and purpose. The principles are the foundation for managing risk and should be considered when establishing the organization's Risk Management Framework and processes. These principles should enable an organization to manage the effects of uncertainty on its objectives.

- **Integrated**: The risk management process (identification of assets, risks and related countermeasures) must be integrated within the organization's activities, otherwise, it would be ineffective.

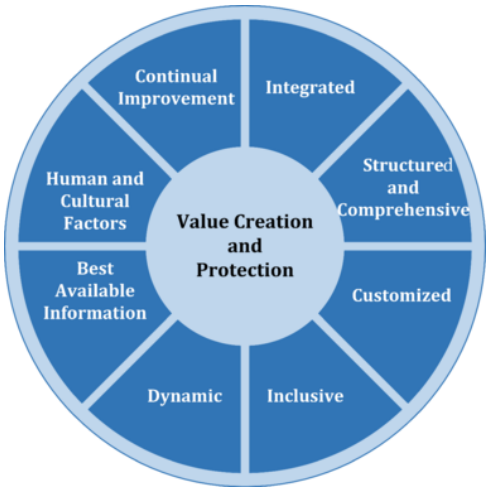


Figure 9: ISO 31000 Principles from [4].

- **Structured and comprehensive**: The risk management process is cyclical and consists of several iterations between the identification of the risk, the application of the treatment, the measurement of the new risk and any new adjustment. With the structuring, we ensure that the results of multiple "measurements" are comparable and with completeness, we guarantee a comprehensive risk analysis.
- **Customized**: Not all organizations are the same and therefore we have to adapt any framework to wer context, internal and external. Otherwise, we conduct a useless analysis and with it also the mitigation of the risk would be in vain.
- **Inclusive**: It is necessary to involve all the parties who play some role in the process (stakeholders etc.) and with whom there must be an exchange of information throughout its duration.
- **Dynamic**: The process must evolve together with the context in which our organization is located (dynamism for the structure). We must ensure that these changes occur in harmony with the outcome of the risk measurements (dynamism over time). If we conduct the analysis once a week, but we have the outcome of a single analysis after a month, we will change something even if we don't need it within the process.
- **Best Available Information**: The inputs to risk management are based on historical and current information, as well as on future expectations. Risk management explicitly takes into account any limitations and uncertainties associated with such information and expectations. Information should be timely, clear and available to relevant stakeholders.
- **Human and Cultural Factors**: This is due to the people involved in the process. It must be remembered that each person reacts to situations, based on their culture and so on. The human factor introduces a bias into the system.
- **Continual Improvement**: Risk management is continually improved through learning and experience.

4.2.2 Framework

The **purpose** of the Framework is to provide support to the organization with a structured process. Without it, every company would have its own way and could go against good practices. **Effectiveness** is measured based on how much the process is integrated within the business dynamics. The more it is integrated with the processes, the more effective it will be.
The phases in which the framework is outlined are the two keywords underlying them: leadership and commitment in Figure 10. The structure of the organization must ensure that roles and responsibilities are outlined and in this case, we need a leader for the process (who synchronizes the various elements of the process itself). Furthermore, all the people involved in the process of identifying assets, risks, countermeasures, etc. they must make a serious commitment to providing the best information to achieve the goal for which they are cooperating. If one of the two elements is lost, the risk management process will not be effective.

Figure 10: ISO 31000 Framework from [4].

- **Design**: how the project should be carried out.
- **Implementation**: materialize the project, concretely design all the elements that are needed for running the process in practice.
- **Evaluation**: observing how the process is behaving.
- **Improvement**: look for improvement.
- **Integration**: integrate the lessons learned, gathering all the experiences and integrate them with the previous project.

Customization is needed and iterating through this life cycle will tend to a better iterating-on-iteration process.

4.2.3 Process

The Risk Management Process involves the systematic application of policies, procedures and practices to the activities of communicating and consulting, establishing the context and assessing, treating, monitoring, reviewing, recording and reporting risk.
This process is illustrated in Figure 11:

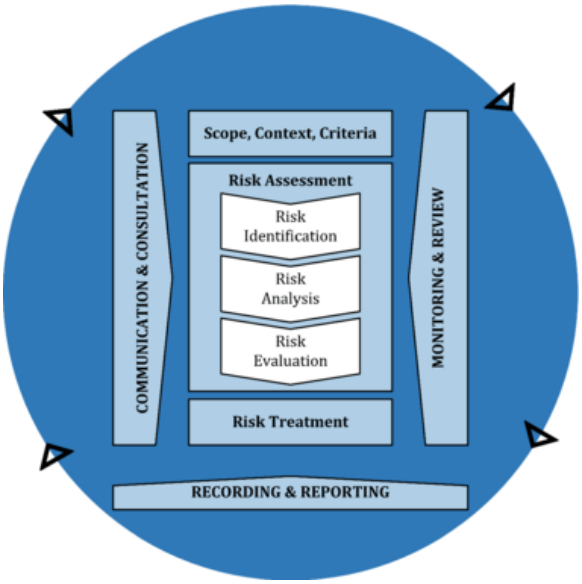


Figure 11: ISO 31000 Process from [4].

The main processes are:

- **Communication and Consultation**: It is a vertical sub-process that supports all risk management. It is a process that manages the correct collection of information and its dissemination among the actors involved in the risk management process. This is the reason for its verticality. It consists of 4 main activities:
  - **Establish a Consultative Team**: Identification of all the people should be involved in the analysis and his formation. This process will provide input to risk management and the Consultative Team doesn't need to be involved in every phase, it is important to just know who we should contact when needed. The composition of the team and its size will be commensurate with that of the company. Is fundamental to give roles and responsibilities of the team members must be clearly defined and specified.
  - **Define a Plan for Communication and Consultation**: We define a way to disseminate information and specify if the communication must be formal or informal, if there are reports that will be propagated, how to obtain information from external stakeholders.
  - **Ensure Endorsement of the Risk Management Process**: Top management must approve everything and a document must be drawn up to that effect. Furthermore, there is a need to reach a common agreement and a mutual compression of how the risk is to be managed.
  - **Communicate Risk Assessment Results**: Important for the principle of awareness. We must not only disseminate information about a risk but also about its possible causes, possible countermeasures. In doing so, everyone has an idea of how they should move.
- **Risk Assessment**: The risk assessment process is divided in 5 steps:
  1. **Context Establishment**: is the identification and documentation of the context relevant for the assessment. We need to focus on the environment in which we are working, distinguishing external and internal elements:
     - **Internal Context**: is related to everything that characterizes and describe the organization from the inside.
     - **External Context**: is needed to consider the regulations that impose constraints specific implementation. Enforces constraints on the design and the possible mitigation that can be applied.

     Involving all the people that can provide a contribution to this activity, namely contact the Consultative Team that will identify which are the relevant stakeholder and support the communication between. We can characterize the Context Establishment in sub-activities:
     - The **target** of assessment is the parts and aspects of the system that are the core of the risk assessment.
     - The **scope** of the assessment is the extent or range of a risk assessment, for the scalability purpose we define what is held inside of and what is held outside of the assessment.
     - The **focus** of the assessment is the main issue or central area of attention in a risk assessment. Identifying the focus is a strict subset of the identification of the scope, but the vice-versa is not true.

     The key elements underlying the context establishment are the assumptions or those preconditions defined a priori. If these are no longer satisfied, then the whole process turns out to be useless if not harmful. Inside the document produced during the Context Establishment phase we should list:

     (a) Identification of the party involved in the Assessment: depending on the party the weight of the risk could be different.
     (b) Assumption declaration.
     (c) Relevant Assets identification for the Risk Assessment.
     (d) Risk scales Definition for consequences and likelihoods, they can be quantitative or qualitative, continuous, discrete, or given as intervals.

(e) Risk evaluation criteria, so terms of reference by which the significance of risk is assessed.

2. **Risk Identification**: The risk identification is the set of activities aiming to identify, describe, and document risks and possible causes of risk. A risk is always associated with an incident. For every real risk we have three essential characteristics:
   - **Asset**: Without assets there is nothing to harm.
   - **Vulnerability**: Without vulnerabilities there is no way to cause harm.
   - **Threat**: Without threats there are no causes of harm.

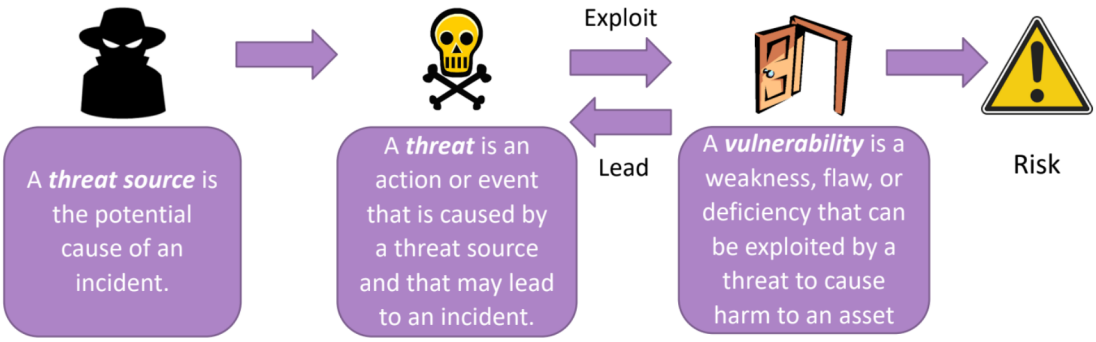


Figure 12: Identifying the risks concerning an asset means identifying threats and how they can lead to an accident through the exploit of a vulnerability. A vulnerability is known to the defender, if not resolved, sooner or later will lead to the attempted exploit of an attacker for which between vulnerability and threat we have the double arrow.

3. **Risk Analysis**: The risk analysis is the activity aiming to estimate and determine the level of the identified risks. The risk level is derived from the combination of the likelihood and consequence:
   - **Likelihood**: is not an easy task, these kinds of events are difficult to model from a statistical point of view, we can try to use the past, perform a historical analysis to evaluate the probability based on the past. All the types of analysis are based on the Bayesian model, there are different techniques but the common point on them is that the data are not clean (redundancy, incomplete, inaccurate etc.). Whatever the technique, we have to consider which threat sources are most important.
   - **Consequence**: is a quantification of damage that we are evaluating. The consequence estimation should be conducted by a walk-through of all identified incidents and assigning the estimates with the involvement of personnel representing the party or someone who can judge consequences on behalf of the party.
4. **Risk Evaluation**: The risk evaluation is the set of activities involving the comparison of the risk analysis results with the risk evaluation criteria to determine which risks should be considered for treatment. Once we have identified likelihood and consequences, we make the product and calculate the value for that risk. We will then put that value within the risk matrix and see what is to be done with that risk, that is, if we can accept it or have to treat it in some way. However, before moving on to any treatment, we must make some considerations. In fact, we must **consolidate the results of the risk analysis** by focusing on the values about which we are most uncertain, once the results are consolidated we assign a value **evaluating the risk level**. Then we try an **aggregation of risks** combining two risks that have similar characteristics from the point of view of the assets they afflict or from the point of view of the sources of threat, treating them as a single risk. Finally, a **risk grouping** is carried out which consists of grouping risks that have elements in common.
5. **Risk Treatment**: The risk treatment is the set of activities aiming to identify and select means for risk mitigation and reduction. Once the risks, sources, costs have been identified and the risk matrix has been elaborated. Depending on the scale used, we will have risks with a high, medium or low level and we can do two things: accept the risk or mitigate it. There are four main options for risk treatment:
   - **Risk Reduction**: reducing the likelihood and/or consequence of incidents.

  - **Risk Retention**: accept the risk either because it has a low probability or because the consequences are easily amortized.
  - **Risk Avoidance**: remove the causes of the risk. It does not reduce the probability or consequences, but rather removes the risk at its root, making sure that the risk cannot materialize in any way. It is usually used in combo with risk reduction.
  - **Risk Sharing**: transfer part of the risk and typically sign a contract with an insurance that shares the risk.

- **Monitoring and Review**: is a single process composed of two main activities:
  - **Monitoring** is the continual checking, supervising, critically observing, or determining the current status in order to identify deviations from the expected or required status.
  - The **review** activity is to determine the suitability, adequacy, and effectiveness of the risk management process and framework, as well as risks and treatments.

  We will apply monitoring and review to the entire applied framework, to the risk management process, to the identified risks and to the measures taken by the organization to address them. This phase is important to verify that the controls are effective and efficient, to see if there are new risks on the horizon, to analyze and learn something even from negative situations such as incidents, to obtain additional information to deal with future risks.

- **Recording and reporting**: Create a history during the time so we can leverage historical data when we perform the analysis. Input for everything: communication and consultation, monitoring and review.

5 Cyber-risk management

- A **cyberspace** is a collection of interconnected computerized networks, including services, computer systems, embedded processors, and controllers, as well as information in storage or transit.
- A **cyber-system** is set of the system that are built on the top of the cyberspace.
- A **cyber-physical system** is a cyber-system that controls and responds to physical entities through actuators and sensors.

Cybersecurity is the protection of cyber-systems against cyber-threats, which can be of two types:

- **Malicious**: Typically caused to produce harm by an attacker (DoS, Injections, Phishing and similar).
- **Non-Malicious**: Typically they are connected to accidental accidents in the system (errors in the design of the process, of a software and the like). Something that was not intentionally created.

5.1 How does Cybersecurity relate to Information Security?

- **Cybersecurity** concerns protection from threats that use a cyberspace.
- **Information security** is the preservation of Information CIA properties.

From here we can reason on the fact that information is not always managed within cyberspace. We can, in fact, have paper documents that contain confidential information and that we have an interest in keeping safe from external threats that have nothing to do with cyber threats. On the contrary, we may have a network to protect but that has nothing to do with the information (we want to protect the infrastructure).

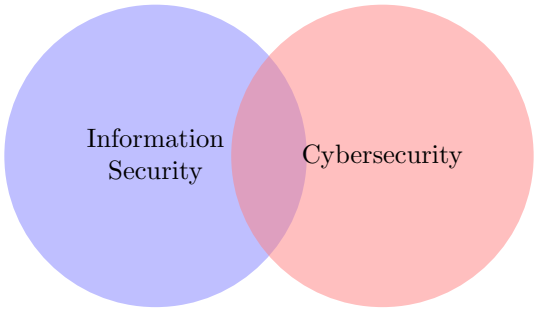


Figure 13: Intersection between Information Security and Cybersecurity.

5.2 How Does Cybersecurity Relate to Critical Infrastructure Protection?

A cyber-physical system acts as a support for so-called critical infrastructures. A critical infrastructure is a particular system that is crucial for the everyday life of citizens: banking systems, healthcare system, electricity and water networks, transport networks. For this reason, Critical Infrastructure Protection (CIP) is one of today's challenges, even more, if possible, the Critical Information Infrastructure Protection (CIIP), or that subset of CIP connected to Information Security. In this field we will find all the activities that aim to support the protection of information systems of critical infrastructures with particular attention to problems related to cyberspace.
As we can see from Figure 14, we have that the protection of critical infrastructures is linked to both Cybersecurity and Information Security, possibly also to both. If we consider the part of CIP that does not intersect with either of the two, we find all the aspects related to the protection of the physical network, but there are also aspects related to Cybersecurity, for example, we want to protect the network that interfaces with the smart grid. The intersection of all three entities encompasses the protection of critical infrastructure information systems that interface with cyberspace. Finally, the intersection between Information Security and CIP concerns all paper documents, patents and the like.

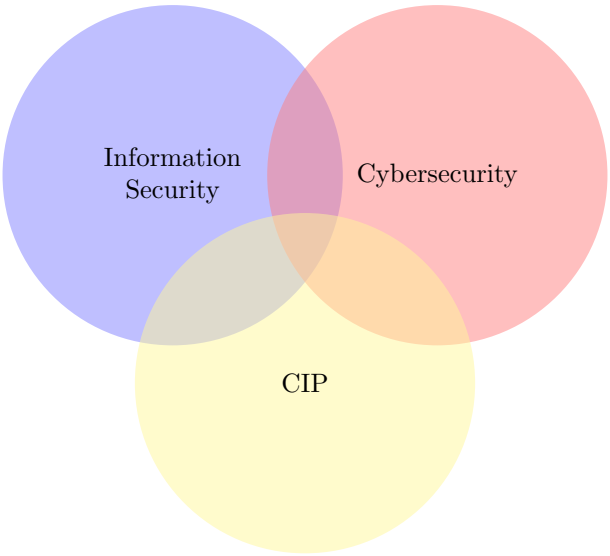


Figure 14: Intersection between Information Security, Cybersecurity and CIP.

5.3 Cyber-Risk

It will therefore make sense to talk about Cyber-risk. As with cyber threats, these particular types of risks can also be classified as malicious and non-malicious. Malicious risks are those that come from malicious threats and so on for others. However, there can be cyber risks that are both malicious and non-malicious, i.e. they can originate from both a malicious threat and a non-malicious one.

5.3.1 Cyber-Risk Assessment

The primary purpose of a cyber-risk assessment is to keep stakeholders informed and support proper responses to identified risks. They also provide an executive summary to help executives and directors make informed decisions about security. We can see it as a procedure consisting of 5 steps:

1. **Context Establishment**
2. **Risk Identification**:
   - Identification of **Malicious** cyber-risks
   - Identification of **Non-Malicious** cyber-risks
3. **Risk Analysis**
4. **Risk Evaluation**
5. **Risk Treatment**

Let's analyze them in detail:

1. **Context Establishment**: Regarding the identification and description of the internal context, however, we must try to determine the so-called attack surface, that is to understand the perimeter of the cyberspace that we are managing and therefore we will have to consider all the possible services that we are exposing to the outside, the points where we get the information from the outside, how we manage it and all the related problems. Furthermore, we have to identify all the assets belonging to cyberspace (servers, services found in the machines), all types of data (database, data in transit, etc.).
2. **Risk Identification**:
   - **Identification of Malicious cyber-risks**:
     we have to take on the role of an attacker and those of the defender who responds step by step to any type of offensive (the risk assessor is the one who watches the game from the outside and tries to anticipate the attacker's moves and his strategies). Empathize with the attacker is very difficult as he can be pushed by the most diverse motivations, he

can have the most disparate computer skills, he can count or not on the help of someone such as institutions or governments and finally he can have at his disposal the most diverse resources possible, from none other than his laptop to sophisticated equipment, etc. Obviously, the risk assessor can rely on a methodology to do his job in the best possible way.

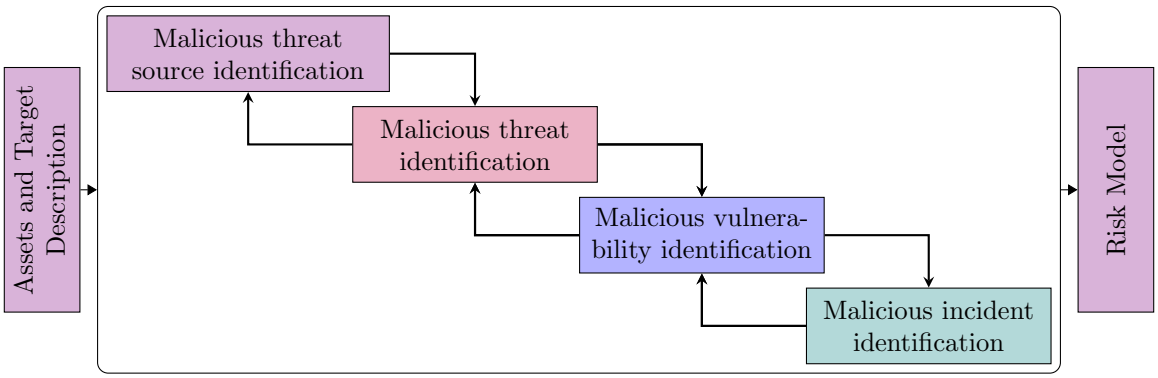


Figure 15: Risk Identification process malicious.

As can be seen from the Figure 15, at first it tries to describe the assets and targets that can be targeted by an attacker, performs 4 specific tasks and then develops a **risk model**. A risk model is simply a quantification that allows you to describe the risk in terms of vulnerabilities, threats and threat sources. The 4 tasks are:

- **Malicious threat source identification**: It is important to understand who may want to initiate attacks, what motivates them, what their capabilities and intentions are and how attacks can be launched. Threat and threat sources are not only external but can also come from inside your cyber-system.
- **Malicious threat identification**: Pay attention to the boundaries of the analysis because attacks can come from both inside and outside. It also documents all the procedures that lead to attacks, with particular emphasis on internal ones although these are unlikely but still possible. Examples of helpful catalogues and repositories that concern cybersecurity (and cyber-threats in particular) are those that are provided by MITRE and OWASP, NIST etc.
- **Malicious vulnerability identification**: Once all threats and threat sources are identified, we come to vulnerabilities. When we analyze vulnerabilities we should focus on the attack surface and see what are the weaknesses of the system. When working with networks, for example, we could use automatic tools to scan the hosts associated with them and detect possible vulnerabilities. As for process-related vulnerabilities, we cannot refer to any tool, but we have to rely on an analyst. Furthermore, we may have vulnerabilities related to human beings (employees who work in the company), for example, lack of awareness of risks or exchange of secret information between members of the different sector organization or even with the outside world. But also access to systems and their management can be a vulnerability: lack of passwords, privilege escalation, etc. All of this is part of the attack surface. To detect vulnerabilities at the application level we can use penetration testing tools and so on.
- **Malicious incident identification**: Having discovered the various vulnerabilities at this point, let's try to understand which accidents can be caused once exploited. Some of the vulnerabilities clearly have different effects than others, so if some exploits affect all 3 security properties of an IT system (Availability, Confidentiality, Integrity) others will only affect two or one. Again, to define incidents we will refer to repositories: data from past experiences shared by other people.

- **Identification of Non-malicious cyber-risks**:
  Here we are not playing any games against any opponent. The incidents that can occur are all involuntary. We will start from the analysis of the incidents to arrive at the threats and sources of threats.

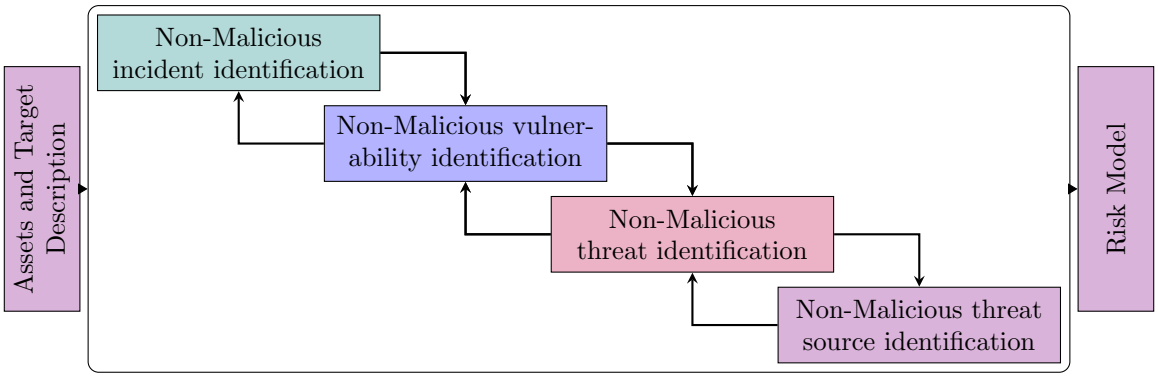


Figure 16: Risk Identification process non-malicious.

We begin to identify the incident, we try to understand which are the vulnerabilities that, if exploited, can lead to that incident. Then we understand what are the possible ways to conduct the exploit, or the threats. And finally, let's see if there is a possible source of non-harmful threat. Generally, the last step is difficult but we will see that in some cases we can use approximations. The 4 steps are:

- **Non-malicious Incident Identification**: Typically they are the result of accidental events whose source is normally given by an external condition. For example, weather conditions, security conditions of the building where the system is located, physical conditions, and so on. So look at what happens outside the organization accidentally. Use logs, previously monitored data and other historical data to facilitate research.
- **Non-malicious vulnerability identification**: In this case, the vulnerabilities are mainly due to some (unintentional) bug in the system. Open sources, such as the ISO 27005 standard, come with lists of typical vulnerabilities.
- **Non-malicious threat identification**: We need to carefully consider the interface to the cyberspace to identify non-malicious threats that arise outside of the system. Relevant sources on typical threats include, for example, the ISO 27005 standard and the NIST risk assessment guide, which provide representative examples of non-malicious threats.
- **Non-malicious threat source identification**: Once we have found the threats we need to find the sources. We can always use the tables offered by ISO/NIST etc, but we could also provide a generic classification of all possible accidental causes: environmental conditions, weather conditions, errors in the system design, errors in one or more phases of the system life cycle, failure of some mechanical component etc. However, historical data and log data help in the search for sources of threat.

3. **Risk Analysis**:
   We remember that in risk analysis we measure the likelihood and the consequences. And when we deal with cyber risks, we have to consider some aspects. Since we have divided the risks into malicious and non-malicious they should also be evaluated differently because malicious risks will have to undergo some considerations while non-malicious ones will have to undergo others. Even if a risk falls into both categories, we need to evaluate it twice.

4. **Risk Evaluation**:
   There are 4 phases at this point of the assessment:

   - **Consolidation of risk analysis results**: Similarly to the general case, focus on the cyber-risks for which the estimates are uncertain and make the distinction between malicious and non-malicious risks.
   - **Evaluation of risk level**: for convenience it is possible to evaluate malicious and non-malicious cyber-risks separately.
   - **Risk aggregation**: equal to the case of general risk. Also, in this case, we check for non-malicious and malicious risks at the same time.
   - **Risk grouping**: Initially consider the two risk entities on their own. And so we try to group non-malicious and then malicious risks. Then we try to consider the risks together.

And therefore we try to group malicious and non-malicious risks indiscriminately. We typically adopt the latter approach when we benefit in both situations with the same treatment.

5. **Risk Treatment**:
   The treatment options are the same as already discussed in the previous chapter: Reduction, Retention, Avoidance and Sharing. In the case of cyberspace, however, most risks can and must be mitigated, due to the technical nature of the threats and vulnerabilities. Typically we will only be able to reduce the likelihood of a certain risk materializing and not much can be done about the consequences. However, we maintain the distinction between malicious and non-malicious risk as the latter can be easily mitigated through environmental interventions, which is not usually possible for malicious risks.

5.4 Useful repositories

5.4.1 MITRE CAPEC

CAPEC is a publicly available catalogue of attack patterns along with a comprehensive schema and classification taxonomy created to assist in the building of secure software. CAPEC helps by providing a comprehensive dictionary of known patterns of attack employed by adversaries to exploit known weaknesses in cyber-enabled capabilities [5].

5.4.2 MITRE CWE

CWE is a community-developed list of common software security weaknesses. It serves as a common language, a measuring stick for software security tools, and as a baseline for weakness identification, mitigation, and prevention efforts [6]. Mitre Corporation maintains a list of disclosed vulnerabilities in a system called Common Vulnerabilities and Exposures (CVE), where vulnerabilities are classified (scored) using Common Vulnerability Scoring System (CVSS) which is a scoring system used to help in prioritizing vulnerabilities' fixing and patching, and it is composed of three metric groups: Base, Temporal, and Environmental. NIST collects and makes available scored vulnerabilities rough the NVD data-base.

6 Risk Management: Challenges and guidelines

6.1 Challenges in the Cyber-Risk Management process

The main difficulties in the Risk Management process are various but the most tedious ones are:

- Which measure of Risk Level to use?
- What scales are best suited under what conditions?
- How to deal with uncertainty?
- High-consequence Risk with low likelihood

6.1.1 Which measure of Risk Level to use?

We have measured the risk level of incidents in terms of consequence for assets and likelihood of occurrence. In other words, we have measured risk level based on two factors, namely loss of asset value when a potential incident occurs and how often this happens.

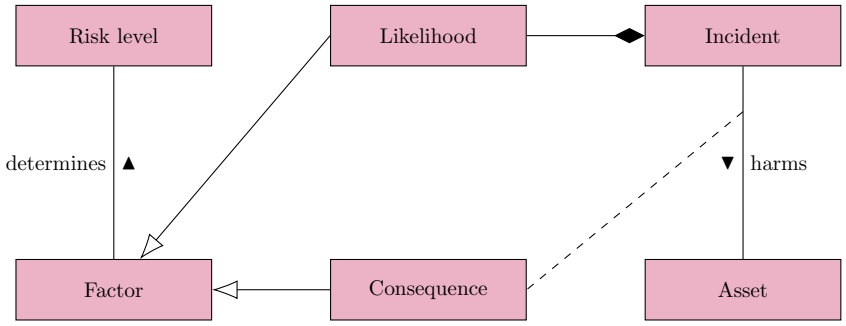


Figure 17: A classical two-factor measure of risk.

From the Figure 17, we actually see how consequences and likelihood are the two factors that determine the level of risk. We also note another thing: although the likelihood is directly linked to the incident, the consequences are directly linked to the relationship between asset and incident, for this reason, the same incident can impact two different assets in a different way. Each line connecting two nodes represents a relation. The white-headed arrows pointing from likelihood and consequence to factor imply that the concepts likelihood and consequence should be understood as instances of the more general concept factor. In other words, they are both factors. Moreover, the factors determine the risk level.

6.1.1.1 Three-factor Measure

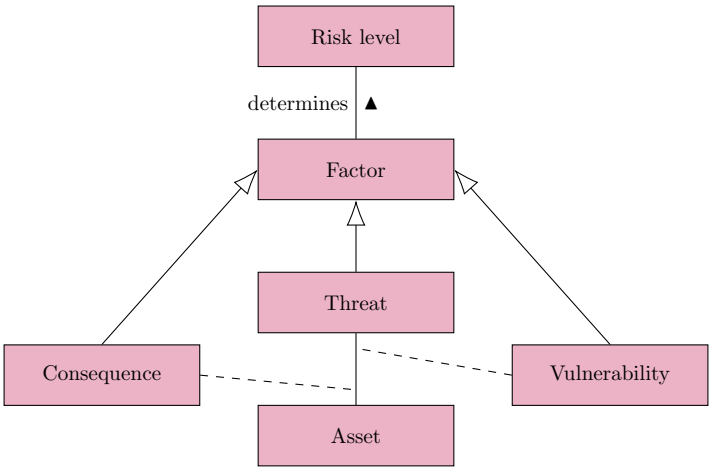


Figure 18: A classical three-factor measure of risk.

This measure defines the concept of likelihood not as a whole but divided between the likelihood that a certain event occurs and the likelihood that the occurrence of the event negatively affects the value of the asset in a tangible way. Likelihood that a threat will occur and likelihood that a threat occurrence will result in an adverse impact. When we measure the likelihood of the event occurring, we take into consideration how much the organization is subject to the event in question. Instead, on the other hand, when we measure that a certain event harms the asset, we take into account the vulnerability that the asset itself exposes to the threat. Unlike the Figure 17, in the Figure 18 we have 3 different factors: we divide likelihood into a threat directly linked to the asset, and vulnerability is linked to the relationship between the asset and the threat.

6.1.1.2 Many-factor Measure

Typical of the OWASP risk rating methodology, this assessment measures the level of risk based on the weighted sum of 4 factors. Two measures for likelihood:

- **Threat agent**: Who could be the source of the attack (skills, motivations, etc.)?
- **Vulnerability**: What are the subsets whose subsets could create problems for the system?

Two measures for the consequences:

- **Technical impact**: Relating to the functioning of the system (availability, integrity and confidentiality of the system).
- **Business impact**: How will compromise the system from a technical point of view impact the organization's business processes/operations?

6.1.1.3 Which measure to use for Cyber-risk?

It depends on the extent of the data we have available during the analysis. In fact, if we don't have historical data, we don't have data extracted from the environment, then a multi-factor approach is very difficult. Before proceeding with the choice of the risk model, we must analyze the data we have available. Therefore, if we can only obtain aggregate data, which has no information about vulnerabilities and threats, then we will have to opt for a two-factor model.
The data collection process is fundamental for the entire risk management process, but data collection tools do not always have an easy life, especially in the world of technology where everything is rapidly changing: just think of an IoT system, in which we don't have always the same set of connected devices and not always, those that fixedly are part of the system are always connected. Furthermore, we often find ourselves obtaining some data from the stakeholders, we must be sure that the information contained in them is reliable and consistent with my risk model and we must analyze by first separating the phase in which we involve this data from the phase in which we use our data.

6.1.2 What scales are best suited under which conditions?

The suitability of a scale depends on:

- the factor in question;
- the kind of risk assessment we are conducting;
- the target of assessment;
- the available data sources.

The possible scales that people can use are:

1. **Nominal Scale** is a discrete scale, where there is subdivision into distinct and mutually exclusive categories. Allow to the same elements the same category, for different elements different categories.
2. **Ordinal Scale**: Same as the nominal scale, where there are disjoint categories, but in this case there is also an order between the categories.
3. **Difference Scale** is an ordinal scale such that equality of difference at the level of values implies that the corresponding phenomena are equally distinct. Allow the determination of equality of intervals or differences.

4. **Ratio Scale** allow the determination of equality of ratios. It is a difference scale such that equality of ratios at the level of values implies that the corresponding phenomena are equally distinct.

Scales 1 and 2 are qualitative while scales 3 and 4 are quantitative. Typically we use quantitative scales when the things to be measured are homogeneous, that is when there is not much difference between the aspects we want to capture, in particular, it is a scale whose use is recommended when the assessment is at a very technical level and requires granularity. On the contrary, we will use the qualitative scale if the things to be measured are not homogeneous and therefore we want to quantify aspects that are different from each other. Moving from a quantitative to a qualitative scale is simple. But, on the contrary, moving from a qualitative to a quantitative scale is very difficult because usually, a qualitative measure is not very expandable.

6.1.2.1 Scales for likelihood

Measuring likelihood means measuring the probability of a certain event happening. Doing this is not easy because in the cyber domain we are subject to many uncertainties and therefore the measures we take are affected by errors. The scale system we choose must be robust concerning errors, data gaps, etc. Most of the time, for likelihood we will use qualitative scales, which more easily absorb uncertainties. However, we could also use a quantitative scale.

6.1.2.2 Scales for consequences

In this case, we cannot make a fixed choice but we have to think about the type of asset we are considering. If we had a perfect mapping between the asset and the cost of its damage then we can use a quantitative scale. But it is also possible to have a hybrid case where we can easily use a quantitative scale and move to a qualitative one with ease. However, for an asset as sensitive as reputation, it is impossible to use a quantitative scale and we will have to use a qualitative scale, as it would be very difficult to map the reputation to the money lost.

6.1.2.3 What scales to use for Cyber-risk?

It depends on the data, the system and the assessor. There is no easy answer also due to the very vast nature of cyberspace and also because there is still a very random factor such as the interaction with the human being. However, if we have a system for automatic data collection, etc., we can venture into a quantitative analysis, otherwise, if we only have one analyst who has to do the work of IDS, etc., we will have to opt for a qualitative analysis.

6.1.3 How to deal with uncertainty?

Uncertainties are one of the reasons for choosing one scale over another. ISO 31000 defines uncertainty to be "the state, even partial, of deficiency of information related to, understanding or knowledge of an event, its consequence, or likelihood". We now distinguish two types of uncertainty:

- **Epistemic**: Uncertainty due to the lack of information, not only due to the quantity, but also to the not excellent quality. It is the most common type of uncertainty that we will have to try to mitigate during the assessment.
- **Aleatory**: Uncertainty due to the randomness of the process. It is a difficult amount of error to calculate and mitigate, usually assumed to be there during the calculations and that's it.

6.1.3.1 How to manage epistemic uncertainty?

We have two options:

1. Using a **quantitative scale** we can represent the uncertainty in intervals. The width of this interval will be directly proportional to the extent of the uncertainty. The amount of tolerable uncertainty depends on the system we are operating on and how it impacts the decision-making process.

		Likelihood				
		Rare	Unlikely	Possible	Likely	Certain
Consequence	Critical					
	Major					
	Moderate					
	Minor					
	Insignificant					

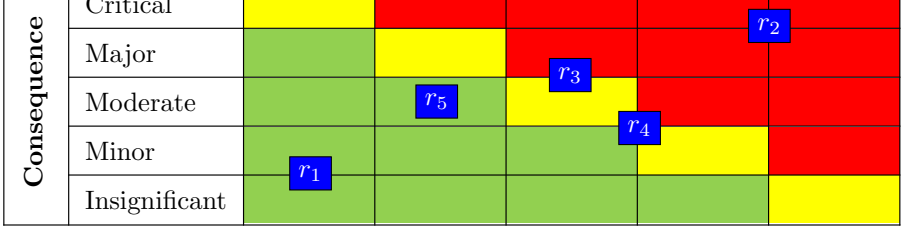


Table 3: Intersection matrix between likelihood and consequences.

Analyzing Table 3 in the case of r_1 and r_2 the uncertainty always leaves the risk in the same region, respectively the green area (low risk) and the red area (high risk) so in this case, the uncertainty is tolerated. We have another situation for r_3: we are sure of the likelihood result but not of the consequences which, due to uncertainty, can be major or moderate. In this case, the uncertainty changes the risk assessment and possibly the risk treatment as if r_3 were medium (yellow zone), so we could somehow tolerate it. If, on the other hand, it fell in the red zone, we would almost certainly be forced to take a countermeasure. We still have the worst case in r_4 as due to the uncertainty (which affects both likelihood and consequences) the risk potentially affects all three colored areas and therefore we have to reduce the uncertainty in a much more prudent way than in the case of r_3. In the case of r_5, on the other hand, we have uncertainty but this does not shift the risk from one class to another, neither in terms of likelihood nor in terms of consequences.

2. Using a **qualitative scale** we can indicate uncertainty as an attribute of the incident.

Incident	Consequence	Uncertainty
Information leakage	Low	Some
Breakdown of server	High	No
Identity theft	High	Considerable
Spyware installed	Medium	Some

Table 4: Consequence of incidents with uncertainty estimates.

To reduce the uncertainties, if needed, we can use different approaches, which however depend on the type of scale we used for the measurements:

- **Fuzzy logic**.
- **Iterating data collection** Useful when data collection for risk calculation is carried out automatically.
- **Comparative Analysis**.
- **Testing** the risk model against historical data or by conducting various surveys across larger groups of stakeholders.

6.1.4 High-consequence Risk with low likelihood

We now focus on a particular risk category: **swan**. In particular, we distinguish two categories of risk:

- A **black swan** is an incident that is extremely rare and unexpected, but has very significant consequences. Typically we associate a medium risk level with such a risk (yellow area in Table 3) as these events are most often connected to external events (such as a threat agent whose likelihood is very low). Medium-level risks are usually not at the top of the risk treatment list and are not always treated, but in this case there is a need for risk treatment. A contingency plan is therefore often associated with black swans.

- A **gray swan** is an incident which has far-reaching consequences, but, unlike a black swan, can be anticipated to a certain degree. These are very neglected types of events and in most cases, they are not presented to the assessor in the input documents, but arise precisely during the analysis. An example in the cyberspace of the gray swan would be a 0-day. We can also try to do something from the point of view of communication: we communicate the risk to management and try to explain it through an appropriate communication strategy based on qualitative scales. The final decision rests with the management who must decide how many resources to dedicate to such a particular risk. If the cost of the hypothetical treatment is high, most of the time there will be no treatment, while if the cost is low, most of the time the management tells you to take action to reduce the risk.

7 Risk Management Methodologies Review

7.1 OWASP Risk Rating Methodology

The OWASP [7] approach is based on the standard two-factors risk model:
$Risk = Likelihood \cdot Impact$. It is composed of 6 phases:

1. **Identifying Risks**
   We collect information on threat agents, which would be the attackers, the attack to be conducted, the exploited vulnerability and, finally, the impact on the business process once the attack is successful.

2. **Factors for estimating likelihood**
   Several factors can help us estimate likelihood, organized into two distinct sets:

   - **Factors for threat agents**: thanks to which we want to calculate the probability that an attack will be successful if performed by a particular threat agent. The numbers inside the brackets are a suggestion for assigning the value based on the description that precedes it. To identify this likelihood we use the following factors that describe the attacker:

Factor	Question	Score
Skill level	How technically skilled is this group of threat agents?	No technical skills (1), some technical skills (3), advanced computer user (5), network and programming skills (6), security penetration skills (9)
Motive	How motivated is this group of threat agents to find and exploit this vulnerability?	Low or no reward (1), possible reward (4), high reward (9)
Opportunity	What resources and opportunities are required for this group of threat agents to find and exploit this vulnerability?	Full access or expensive resources required (0), special access or resources required (4), some access or resources required (7), no access or resources required (9)
Size	How large is this group of threat agents?	Developers (2), system administrators (2), intranet users (4), partners (5), authenticated users (6), anonymous Internet users (9)

Table 5: OWASP factors for threat agents.

   - **Factors for vulnerability**: with these factors, we calculate the likelihood of a vulnerability based on certain characteristics:

Factor	Question	Score
Ease of discovery	How easy is it for this group of threat agents to discover this vulnerability?	Practically impossible (1), difficult (3), easy (7), automated tools available (9)
Ease of exploit	How easy is it for this group of threat agents to actually exploit this vulnerability?	Theoretical (1), difficult (3), easy (5), automated tools available (9)
Awareness	How well known is this vulnerability to this group of threat agents?	Unknown (1), hidden (4), obvious (6), public knowledge (9)
Intrusion detection	How likely is an exploit to be detected?	Active detection in application (1), logged and reviewed (3), logged without review (8), not logged (9)

Table 6: OWASP factors for vulnerability.

3. **Factor for estimating impact**
   When considering the consequences of an attack it is important to realize that there are two types of impact:

   - **Technical impact**: affecting applications and data mainly. The technical impact is divided into:

Factor	Question	Score
Loss of confidentiality	How much data could be disclosed and how sensitive is it?	Minimal non-sensitive data disclosed (2), minimal critical data disclosed OR extensive non-sensitive data disclosed (6), extensive critical data disclosed (7), all data disclosed (9)
Loss of integrity	How much data could be corrupted and how damaged is it?	Minimal slightly corrupt data (1), minimal seriously corrupt data (3), extensive slightly corrupt data (5), extensive seriously corrupt data (7), all data totally corrupt (9)
Loss of availability	How much service could be lost and how vital is it?	Minimal secondary services interrupted (1), minimal primary services interrupted OR extensive secondary services interrupted (5), extensive primary services interrupted (7), all services completely lost (9)
Loss of accountability	Are the threat agents' actions traceable to an individual?	Fully traceable (1), possibly traceable (7), completely anonymous (9)

   - **Business impact**: it is the most important and concerns the economic and legal fabric of the attacked organization. The impact on the business is divided into:

Factor	Question	Score
Financial damage	How much financial damage will result from an exploit?	Less than the cost to fix the vulnerability (1), minor effect on annual profit (3), significant effect on annual profit (7), bankruptcy (9)
Reputation damage	Would an exploit result in reputation damage that would harm the business?	Minimal damage (1), Loss of major accounts (4), loss of goodwill (5), brand damage (9)
Non-compliance	How much exposure does non-compliance introduce?	Minor violation (2), clear violation (5), high profile violation (7)
Privacy violation	How much personally identifiable information could be disclosed?	One individual (3), hundreds of people (5), thousands of people (7), millions of people (9)

4. **Determining the Severity of the Risk**
   Here we put together the estimates of likelihood and consequences to calculate the severity of the risk. We generally rely only on technical impact if the business impact estimate is not good. Typically the mapping between the numerical value and risk level is like this:

Likelihood and Impact Levels	
0 to < 3	LOW
3 to < 6	MEDIUM
6 to 9	HIGH

   However, we can conduct this analysis by following two approaches:

   - **Informal Method**: In some cases, it is not wrong to consider looking only at the summary values of some factors and simply deducing the most important aspects of the risk analysis through some considerations. Identify and focus on the key factors only.

- **Repeatable Method**: If, on the other hand, it is necessary to make the assessments repeatable, we must resort to a more rigorous process of calculating the risk factors. The concept of uncertainty is always valid in this case. These estimates are intended as an aid to the assessor to arrive at a reasonable result.

Very often, however, automatic tools are used to calculate these factors. An example of a repeatable method is the following:

Threat agent factors				**Vulnerability factors**			
Skill level	Motive	Opportunity	Size	Ease of discovery	Ease of exploit	Awareness	Intrusion detection
5	2	7	1	3	6	9	2
Overall likelihood = 4.375 (MEDIUM)							

Technical Impact				**Business Impact**			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability	Financial damage	Reputation damage	Non-compliance	Privacy violation
9	7	5	8	1	2	1	5
Overall technical impact = 7.25 (HIGH)				Overall business impact = 2.25 (LOW)			

Here we ask ourselves if the information related to business impact is good. If so, we calculate the business impact, otherwise we use the technical impact. Once the risk level has been calculated, we insert it into the risk matrix Table 7 and observe that:

Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	None	Low	Medium
		LOW	MEDIUM	HIGH
		Likelihood		

Table 7: Overall Risk Severity.

Considering the business impact, the overall risk severity is low, while if we considered the technical impact, the overall would be high. The context of vulnerabilities therefore becomes crucial at this stage.

5. **Decide what to fix**: Once the overall risk level has been established, the risks to be fixed must be prioritized. The first rule of thumb is: higher-level risks are mitigated first. Don't focus on the little things that cost little first. The second general rule is: not all risk fixes are smart, something that gives me little improvement and costs a lot, is not worth it.
6. **Customizing Risk Rating Model**: Having a customizable framework for a company is essential for it to use it. There are several ways to develop a custom model and among them stand out:
   - **Adding Factors**: the assessor can decide whether to choose different factors to represent what is important for the company.
   - **Customizing options**.
   - **Weighting factors**: Earlier we assumed that all factors are equally important, but during the customization process, we can make the factors most important to our business matter more in some way and so we will have to make a weighted average in the calculation of the likelihood and consequences.

The **pros** are: structured in few simple steps, is easy to apply and is supported by open source tools. The **cons** are: allows only a course grained risk analysis and does not provide too much support to the risk mitigation phase.

7.2 CRAMM - CCTA Risk Analysis and Management Method

This is a qualitative risk analysis tool developed by the Central Computer and Telecommunications Agency (CCTA), whose purpose was to provide government departments with a method for reviewing information security [8]. Computer Risk Analysis and Management Method (CRAMM) is based on meetings, interviews and structured questionnaires for data collection. The tool includes 3 stages, each supported by questionnaires and guidelines, the first two identify and analyze the risk, while the last recommends how to manage it:

1. **Identification and Valuation of Assets**:
   Three types of assets are identified: data, application software and physical assets.

2. **Threat and Vulnerability Assessment**:
   Threats and vulnerabilities are examined based on asset groups. The tool offers us a series of tables for threat/asset group and threat/impact combinations. All it wants to do is provide a managerial risk assessment, so, too technical information about the vulnerabilities that can be detected by scanners is not used by CRAMM. However, to assess threats and vulnerabilities we can follow two approaches:

   - **Full risk-assessment**: The analysis of vulnerabilities and threats is conducted through questionnaires that allow the tool to receive all the answers, which, based on what it has obtained, calculates the level of threat per asset group and the level of vulnerability for each threat.
   - **Rapid risk-assessment**: Threat and vulnerability levels are entered directly into the tool via a classification guide.

3. **Identify and prioritize countermeasures**:
   At this point, CRAMM produces a series of countermeasures applicable to the system. The recommended safety profile is then compared to the one in force and areas that need to be strengthened or areas that need a softening of safety measures are identified. The strength of CRAMM is given by the help it offers to the assessor during the prioritization of countermeasures. In particular, these will have higher priority if:

   - it protects against several threats;
   - it is required to protect a high risk system;
   - there are no alternative countermeasures already installed;
   - it is less inexpensive to implement (based on a general cost estimation);
   - it is more effective to meet the objectives of its sub-group;
   - it prevents an incident rather than detect or facilitate recovery.

The **pros** are: structured in few simple steps and is oriented to the management.
The **cons** is that it can be only applied by using a proprietary tool.

7.3 MEHARI Methodology

MEHARI (MEthod for Harmonized Analysis of RIsk) is an open-source method for risk assessment and analysis. MEHARI is designed to be ISO 27005 compliant and in particular in the context of an ISO 27001 certified organization [9]. The methodology is divided into 3 phases: **Risk Assessment**, **Risk Treatment** and **Risk Management**. As can be seen from Figure 19, in the first phase we have a further subdivision that involves the identification, estimation and assessment of risk.

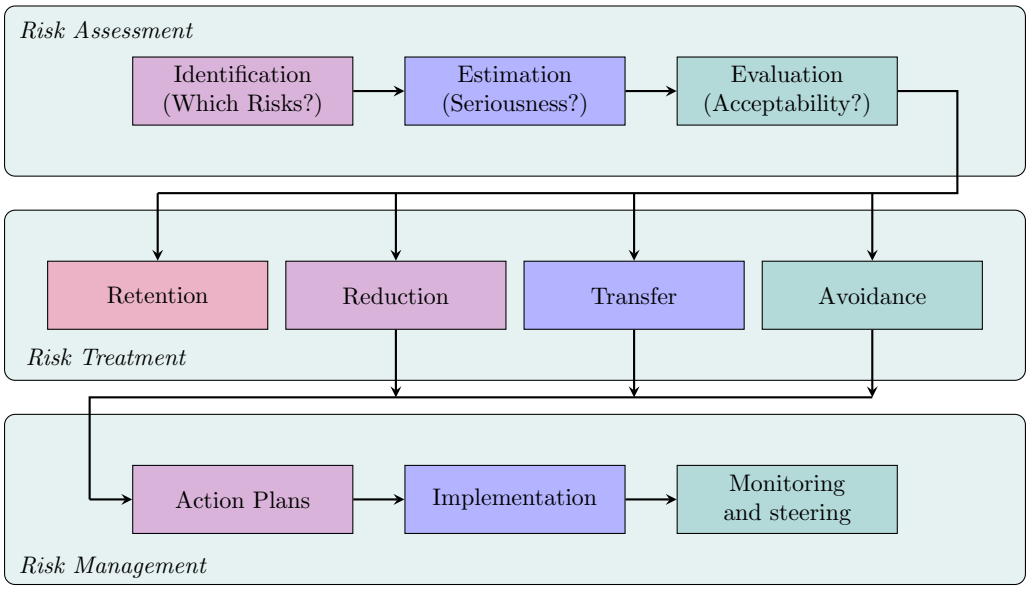


Figure 19: MEHARI Methodology.

7.3.1 Risk Assessment

Let's now see the Risk Assessment phase in detail.

7.3.1.1 Identification

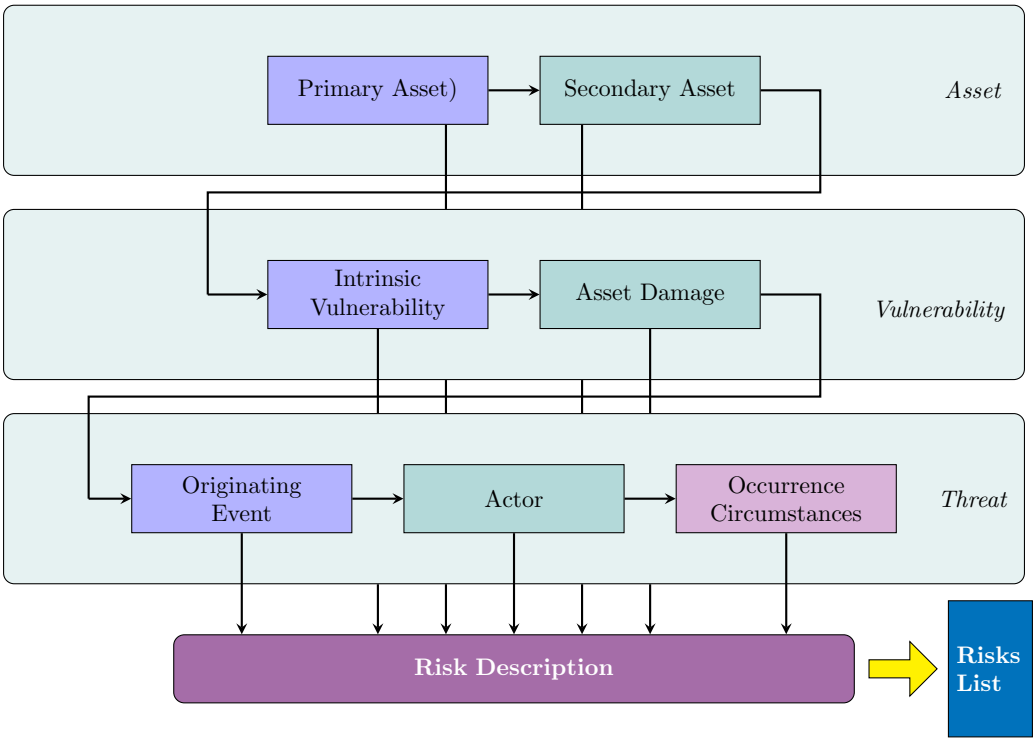


Figure 20: MEHARI Identify Risks.

At this stage we ask ourselves two questions:

1. What characteristics of risk do we need to emphasize and how detailed do we need to go into each of them?
2. What is the best way to do this?

Let's analyze the risk like in Figure 20:

1. **Asset**
   As we have already extensively seen, assets are the protagonists of a risk analysis. The consequences and the severity of a certain risk are connected with the nature of the asset. We now distinguish two types of assets:
   - **Primary asset** shall be described according to the following Table 8:

		Asset Categories		
		Services (IT and General)	Data necessary for the services to function	Management Processes
Asset classification criteria	Types of needs			
	Type of service providers			
	Field of activity and different areas of responsibilities			
	Technology used			
	Users concerned			

Table 8: This table is used for documentation purposes.

   - **Secondary asset** shall be described by types of means required to meet the functional needs described by the primary assets.
2. **Vulnerability**:
   - **Intrinsic Vulnerability**: It is an intrinsic feature of a system, object or asset that may be susceptible to threats.
   - **Contextual Vulnerability**: it is defined as a shortcoming or flaw in a security system that could be exploited by a threat to strike a targeted system, object or asset.
3. **Threat**
   - **Originating Events**: The threat that leads to the exploitation of a vulnerability, in MEHARI, is not trivially attributable to a causal event. Rather, anything can be used to describe how harm can occur and thus affect the likelihood of a risk. In particular, we will pay attention to various factors and among these, we find the triggering event which can be of 3 types, as can be seen in Figure 21:
     - Accidents
     - Errors
     - Voluntary Acts

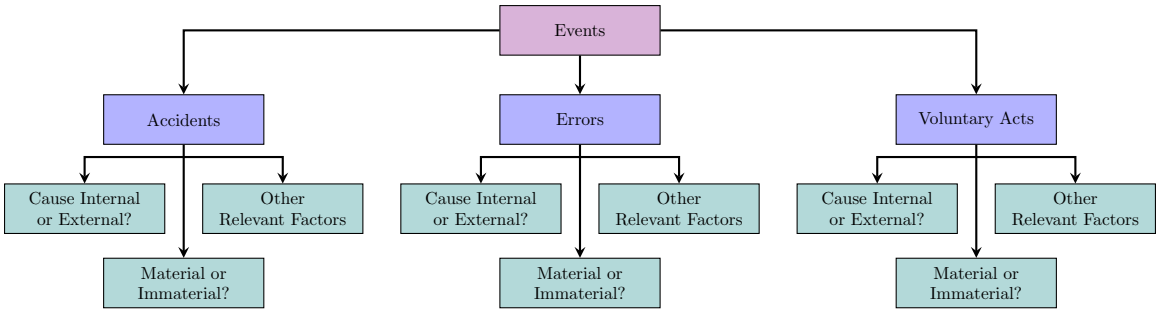


Figure 21: Originating events.

- **Actors**: In the case of threats that are originated by people, it is important to distinguish categories of people according to their rights and privileges. According to these rights:
  - Actors may be more or less capable of originating the threat.
  - The security measures that should be implemented will be different.

Figure 22 summarizes the main stages of Risk Identification.

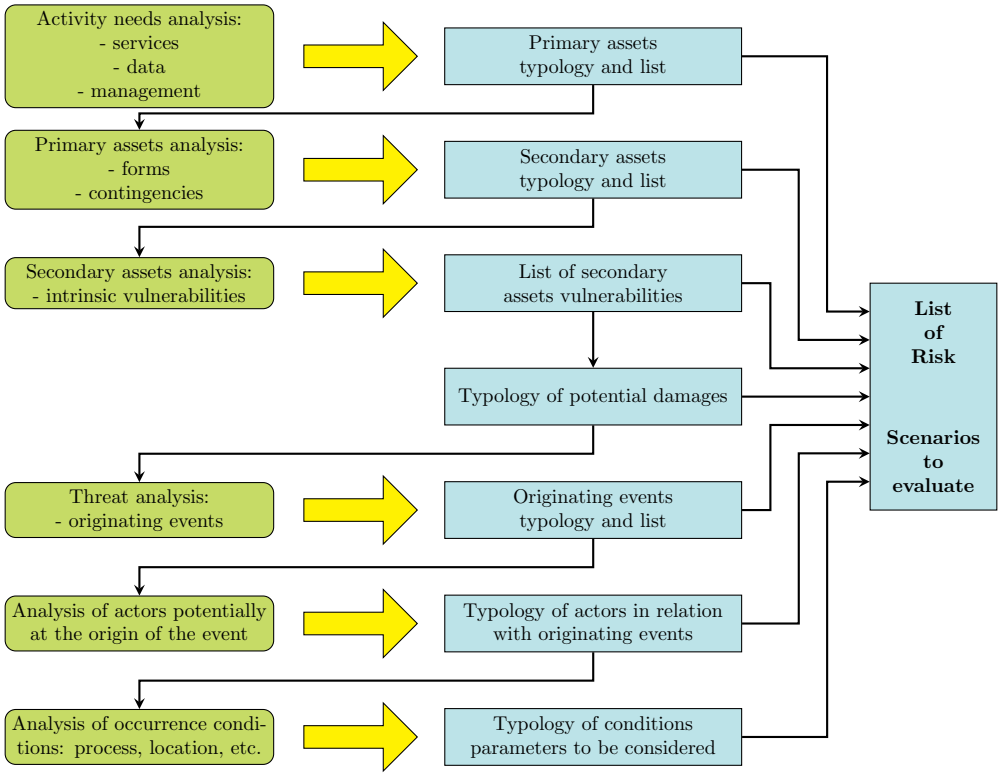


Figure 22: Summary of Risk Identification.

7.3.1.2 Estimating Risks

Here we are concerned with calculating how serious the risk is. To do this, we have said that for each of those found we calculate the overall likelihood, the overall consequences and finally we calculate the level of risk. But usually, a calculation of likelihood and consequences is difficult so we use a more analytical approach that takes into account:

- **Intrinsic impact**: Given the maximum possible impact that the organization must face in the total absence of specific security features to reduce that risk.
- **Intrinsic likelihood**: Given by the maximum likelihood value relating to the concretization of risk in the total absence of security measures that want to contain it.
- The effects of the security measures on the two previous parameters.

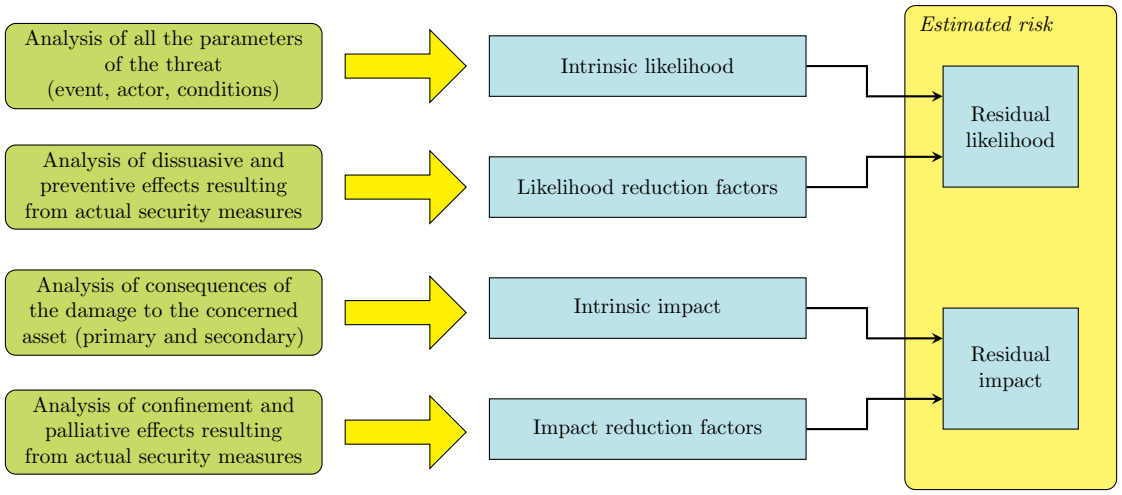


Figure 23: Summary of risk scenario assessment.

The risk estimate would therefore be a summary of two new measures: residual likelihood and residual impact.

7.3.1.3 Evaluating Risks

Is this risk acceptable as it is or do we have to act on it in some way? We will base our answer on the decision table which will guarantee consistency between the decisions taken by different management at different times. At the base of the decision table there will be a certain acceptability criterion thanks to which we can define 3 risk categories:

- Intolerable risks: They require an emergency measure outside the normal budget cycles.
- Inadmissible risks: They must be mitigated or eliminated within a certain time (but without urgency, as in the case above).
- Acceptable risk.

In MEHARI 2010, the acceptability criterion is as in Table 9:

I = 4	S = 2	S = 3	S = 4	S = 4
I = 3	S = 2	S = 3	S = 3	S = 4
I = 2	S = 1	S = 2	S = 2	S = 3
I = 1	S = 1	S = 1	S = 1	S = 1
	L = 1	L = 2	L = 3	L = 4

Table 9: Acceptability table from MEHARI 2010.

S is the global seriousness evaluated as a function of the impact (I) and likelihood (L).

- Level 4: intolerable risks.
- Level 3: inadmissible risk.
- Level 2 and Level 1: acceptable risks.

7.3.2 Risk Treatment

As can be seen from Table 9, for the Risk Treatment we can reassure that if we are in a red area the risk must be treated immediately even if we do not have the budget to mitigate it. If we are in the yellow area we are in an area that tells us to deal with the problem as soon as possible. Finally, if we are in the green area we can also decide to ignore the risk.

7.3.3 Risk Management

The third main block is represented by the activities relating to Risk Management. The first activity is the **Action Plans** within which a subset of all possible mitigation actions is selected, which basically means finding the best trade off between intrinsic probability, probability reduction factors, intrinsic impact and impact reduction factors.
Once we have found the list we have to implement it, the **implementation** is obviously something that is context dependent.
When we have implemented everything in the system we are ready to observe the mitigation results, so we're ready to start to **review** possible analyses that we performed and we can start collecting information to initialize a further iteration of the risk management process, in the end, MEHARI supports the direct control loop that we identified at the beginning of the course.

8 Cyber-risk Management: A Case Study

Our case study is based on a simplified version of a risk assessment conducted on a smart grid infrastructure. Recall that a smart grid is the combination of an information network and an electricity distribution network that allows you to manage the supply of energy between users in an "intelligent" way. The objective of the case study is to assess the risk associated with the information network that supports the monitoring of electricity between producer (power plant, provider) and consumer. The assessment also includes all the components for the on/off energy switch and those related to the limit of electricity that can be supplied (choking). We have identified the assets to be protected and it is useful to assess this risk for a certain party which in this case is the distributor system operator. Cyber-Risk Assessment recap:

1. Context Establishment
2. Risk Identification:
   - Identification of Malicious cyber-risks
   - Identification of Non-malicious cyber-risks
3. Risk Analysis
4. Risk Evaluation
5. Risk Treatment

8.1 Context Establishment

Identification and description of the context, in this case, it is given by the exercise, so we will only have to read and understand the external and internal environment of the organization. Here we interact with the stakeholders. Typically are also defined: goals, objectives, assumptions, assets to be defended, the scale of measurement for likelihood and the consequences and criteria for risk assessment.

8.1.1 External Context

We must describe all relevant aspects (business, legal, regulatory, financial, etc.) imposed on the supplier. It is a smart grid which is a critical infrastructure so the reference stakeholders tell us, for example, that the organization is subject to the **European NIS** directive which we must take into account during the evaluation. In the example, we will have to guarantee a certain availability of the service and the periods of inactivity must have an adequate duration, etc.

8.1.2 Internal Context

We must describe all the relevant aspects related to what the organization does (objectives, policies, capabilities, etc.). In the case in question, the primary objective is the distribution of electricity to customers, but this consists of some sub-goals:

- The supply must be guaranteed in a reliable/safe manner.
- The exchange of information between providers and customers must be stable and timely as these data are the basis of the monitoring process which in turn is the input for the amount of energy to be supplied.
- Protect the privacy of customers.

Another important piece of information is given about the people who work within the organization, many of the operators have strong skills and some staff members have received specific training regarding risk assessment.

8.1.3 Goals and Objectives of the Assessment

1. Conduct the analysis to identify and mitigate, where possible, the risks that would prevent the organization's business process from carrying out.
2. Conduct the analysis compatibly with the laws and regulations to which the organization is subject.
3. We would like the results of the analysis to be disseminated within the company context and to (external) stakeholders to increase awareness of the risks, we must avoid dwelling on technical details.

8.1.4 Target of the Assessment

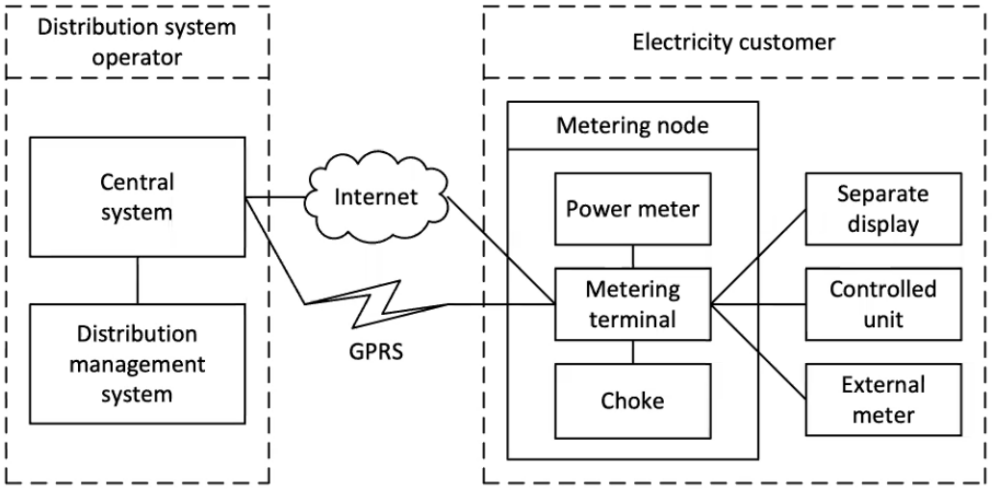


Figure 24: Smart grid system.

The system in the Figure 24 consists of two parts:

- **Relating to the operator of the distribution system**: This is composed of two applications: the central system and the distribution management system. These are run on a machine that is part of a provider-controlled network.
- **Relative to the customer**: Here we have the metering node which would be the smart meter installed at the customer, which controls the amount of energy, how this is used and sends the information to the energy producer, so that this can manage the right energy to send home and understand how much it has to make us spend. The meter is made up of the hardware part, i.e. the sensors or power meters, the metering terminal i.e. the software and finally the choke that determines the physical quantity of incoming and outgoing energy.

8.1.5 Scope of the Assessment

For customers and providers, we have two interfaces with cyberspace: one with the internet and one with the GPRS network. These are the limits within which we can move and which outline the attack surface of the system. In particular, external attackers can use the internet and/or the GPRS network to damage the system. However, we can also identify internal attackers, who are, in most cases, on the customer side. Why is this the case in most cases? Because according to the above premises, we have said that the staff is well trained, the system is well protected, physical access limited to the devices, we can therefore guess that it is on the customer side. We decide to reduce the scope of the assessment, after discussing with the customer, to the analysis of the system interface that faces the Internet, as there is another parallel assessment, exclusive to the interface between the system and the network GPRS. Let's lighten the analysis, even more, taking into account not having to consider back-end attacks.

8.1.6 Focus of the Assessment

1. Exchange of meter data and control data via the Internet and the ways in which this may affect the provisioning of power, as the distribution system operator is particularly concerned about this aspect.
2. Although within the scope, the main focus will not be on attacks via physical access to components.
3. Risks caused by malicious as well as non-malicious threat sources should be considered.
4. Regarding functionality, the focus of the assessment is on basic Advanced Metering Infrastructure (AMI) functions, which include:
   - registering electricity customer meter data;
   - transfer of data between Electricity customer and Distribution system operator;
   - switching on/off or choking of power provided to the electricity customer.

8.1.7 Assumptions

1. Threat sources may be both internal and external.
2. Malicious and non-malicious threats may be both internal and external.
3. The target of assessment may be targeted not only by individuals with a purely financial or personal motive, but also by actors who wish to disrupt society.
4. All meter data and control data sent between the central system and metering nodes are encrypted.

8.1.8 Assets

The distribution system operator is the sole entity for the cyber-risk assessment, which means that all consequence assessments and risk evaluation criteria will be defined from that perspective.

- **Integrity of meter data**: The integrity of meter data should be protected all the way from Power meter to Distribution system operator.
- **Availability of meter data**: Meter data from Metering node should be available for Distribution system operator at all times.
- **Provisioning of power to electricity customers**: Power should only be switched off or choked as a result of legitimate control signals from Central system.

8.1.9 Scales

In this case, the risk model will be of the two-factor type and therefore for the level of risk, we will only have to calculate consequences and likelihood.

- **Likelihood scale**:
  Since the assessment is of a high level and involves a high level of information, we choose a discrete scale. In particular, we will choose a 5-level scale which is defined in terms of frequencies that directly refer to the probabilities of a certain event happening.

Likelihood value	Description
Rare	Less than once per ten years
Unlikely	Less than once per two years
Possible	Less than twice per year
Likely	Two to five times per year
Certain	Five times or more per year

  This scale is used for two reasons: firstly, it is easier to make the manager layer understand the frequency with which certain events occur and, secondly, we can have access to historical data and logs files that allow us to calculate it in an easy way.

- **Consequence Scales**:
  Consequences are related to Assets, so we have a different scale for each asset to be defended. The scale must be elaborated from the point of view of the provider/operator taking into account the damage that the business process may suffer. We have 3 assets and therefore we will have 3 different scales:

  1. Integrity in Table 10:

Consequence value	**Description**
Insignificant	Errors in meter data for up to 100 electricity customers
Minor	Errors in meter data for up to 101-2,000 electricity customers
Moderate	Errors in meter data for up to 2,000-20,000 electricity customers
Major	Errors in meter data for up to 20,001-50,000 electricity customers
Critical	Errors in meter data for more than 50,000 electricity customers

Table 10: Consequence scale for integrity of meter data.

  2. Availability in Table 11:

Consequence value	**Description**
Insignificant	Meter data for up to 1,000 electricity customers unavailable for 1-24 hours
Minor	Meter data for up to 1,000 electricity customers unavailable for more than 1 day or meter data for 1,001-10,000 electricity customers unavailable for 1-24 hours
Moderate	Meter data for up to 1,001-10,000 electricity customers unavailable for more than 1 day or meter data for more than 10,000 electricity customers unavailable for 1-24 hours
Major	Meter data for more than 10,000 electricity customers unavailable for 25 hours-7 days
Critical	Meter data for more than 10,000 electricity customers unavailable for more than 7 days

Table 11: Consequence scale for availability of meter data.

  3. Provisioning in Table 12:

Consequence value	**Description**
Insignificant	Power outage for up to 100 electricity customers for 1-24 hours
Minor	Power outage for up to 100 electricity customers for more than 24 hours or power outage for 101-1,000 electricity customers for 1-24 hours
Moderate	Power outage for 101-1000 electricity customers for more than 24 hours or power outage for 1,001-10,000 electricity customers for 1-24 hours
Major	Power outage for 1,001-10,000 electricity customers for 25-72 hours or power outage for more than 10,000 electricity customers for 1-24 hours
Critical	Power outage for 1,001-10,000 electricity customers for more than 72 hours or power outage for more than 10,000 electricity customers for more than 24 hours

Table 12: Consequence scale for provisioning of power to electricity customers.

8.1.10 Risk Evaluation Criteria

Given the likelihood and consequences scales, we need to define the risk assessment criterion, i.e. the matrix colouring. We assume that each asset has a 5-level consequence scale and the likelihood scale is the same for all three assets, so we use the same metric for all three risks. If we think that the matrix is different for one or more assets than those to be defended, we can create a new matrix in which we leave the axis of likelihood unchanged but we change the scale of the consequences. Normally a metric is used that provides 3 levels of risk (green, yellow and red), where the yellow boxes are on the main diagonal of the matrix (Table 13).

		Likelihood				
		Rare	Unlikely	Possible	Likely	Certain
Consequence	Critical					
	Major					
	Moderate					
	Minor					
	Insignificant					

Table 13: Risk Evaluation Criteria.

8.2 Risk Identification

The goal is to arrive at a collection of:

- Threat sources
- Threats
- Vulnerabilities
- Incidents
- Risks

that is as correct and complete as possible for our particular target of assessment and assets.
Various techniques can be used in this regard:

- Analysis of data from the system: Logs, IDS and other monitoring tools, vulnerability scanners, results of penetration testing and other security tests, router policies, etc.
- External vulnerabilities and external repositories (e.g. NVD database or ISO 27005).
- Interaction with people to validate the system's coded information.

When we use historical data, we keep in mind that it can be a good indication but as for the future, we do not expect a situation quite the same as in the past.

8.2.1 Identification of Malicious Cyber-risk

8.2.1.1 Threat Source Identification

We need to understand who the attackers are and why they want to damage the system. This allows us to quantify the likelihood of a specific incident. Not all attacks will have the same likelihood if the threat source changes. We conduct this research by creating a table with 3 columns:

- The source of threat we want to consider. We can use the banalest classification between internal and external sources, but we can also classify, in a more elaborate way, taking into account the attacker's abilities.
- The motivations and intentions.
- The actual capabilities and resources available.

In the example case, we can consider the following attackers (Table 14):

Threat source	**Motive and intention**	**Capability and resources**
Script kiddie	Achieve status among a group or prove his/her ability to cause harm. Will seldom be very persistent if faced with difficulties and initial failure	Relatively unskilled, unable to perform complicated attacks. Typically uses tools developed by others to initiate attacks. Very limited access to computational or monetary resources.
Cyber-terrorist	Cause disruption in a society thorough cyber-attacks, preferably against critical infrastructure. Strong political, ideological, or religious motives and willingness to go to extremes.	May command significant resources and skill, in some cases even being supported by nation states. Able to perform long-term planning, preparation, and carrying out of attacks.
Black hat hacker	Motivated by personal gain, for example through tampering with data or blackmail. This includes, for example, electricity customers who seek to reduce their electricity bill by tampering with meter data.	The skill level of black hat hackers can vary a lot, but the best are world-leading experts on cybersecurity. If part of a larger criminal organization, they can also command significant resources.
Hacktivist	Similarly to cyber-terrorists, hacktivists are motivated by a political, ideological, or religious agenda and use cyber-attacks to achieve their goals. Although the distinction between cyber-terrorists and hacktivists are less willing to go to extremes and that their aim is to harm selected groups, politicians, or other individuals, rather than society as a whole.	Skill level and resources can vary a lot. Most hacktivists are assumed to operate alone or in small or poorly organized groups. However, if well organized they can potentially have access to significant computational resources as well as competence.
Insider	An insider is a disloyal employee or consultant of the distribution system operator who is typically motivated either by personal gain or by a desire to harm the employer due to conflicts and discontent.	May have access to all systems and possess detailed information and knowledge about the system architecture, functionality, and security features.
Malware	By malware we mean here malicious software developed to harm computerized systems, but which are not aimed specifically to harming the assets of the party of the risk assessment.	Developers of malware are often highly skilled. Malware can therefore cause significant harm to systems based on standard off-the-shelf operating systems or other software.

Table 14: Malicious threat sources.

8.2.1.2 Threat Identification

At this point we also create a table consisting of 3 columns in which we put:

- **Threat source**.
- **Attack point**: which would be the place where the source of the threat can start the attack.
- **Threat**: that is a synthesis of resources, source of threat and point of attack.

For this particular exercise we can have (Table 15):

Threat source	Attack point	Threat
Script kiddie	Internet connection to the central system	DDoS attack on the central system
Cyber-terrorist	Same as the row above	Same as the row above
Cyber-terrorist	Internet connection between the central system and the metering terminal	Tampering with all or most control data in transit from the central system to the choke component
Black hat hacker	Internet connection between the central system and the metering terminal	Tampering with data in transit from the metering terminal to the central system
Black hat hacker	Communication line between the metering terminal and the external meter	Malware to manipulate meter data is installed on the metering terminal through connection to the external meter
Malware	Internet connection to the metering terminal	Metering node infected by malware
Hacktivist	Internet connection between the metering terminal and the central system	Tampering with control data in tranist from the central system to the choke components for selected electricity customers
Insider	Central system	Illegitimate control data sent to the choke components from the central system

Table 15: Malicious threats.

8.2.1.3 Vulnerability identification

For each threat, we identify the vulnerabilities that exist that threaten to create actual harm. We can look at the vulnerability lists offered by NISTIR 7628, ISO 27005, OWASP or CWE. Also we make a table with:

- **Threats**.
- **Vulnerability** found in the system associated with the threat.
- Brief **description** of the vulnerability.

In this case, we are seeing high-level and non-software related vulnerabilities (Table 16).

Threat	Vulnerability	Description
DDoS attack on the central system	Inadequate attack detection and response on central system	New forms of DDoS attacks are continuously being developed to defeat existing countermeasures. Due to the challenges of keeping the central system running 24/7, combined with the lack of a strong tradition for cybersecurity awareness in the power distribution domain (which has not traditionally operated in cyberspace), countermeasures to various forms of DDoS attack on the central system are rarely updated and may therefore be out of date.
Tampering with all or most control data in transit from the central system to the choke component	Weak encryption and integrity check	The encryption of messages between the central system and the metering node may be weak compared to the current standard. The same applies to the integrity checking of received messages. This applies in particular at the metering nodes, which have relatively little computing power and are rarely replaced.
Tampering with control data in transit from the central system to the choke components for selected electricity customers.	Weak encryption and integrity check	The considerations here are the same as in the previous row.

Malware to manipulate meter data is installed on the metering terminal through connection to the external meter.	Unprotected local network, no sanitation of input data from the external meter	The local network at the electricity customer location cannot be assumed to be properly protected, as this depends on the individual customer. Moreover, data from the external meter to the metering terminal are not adequately sanitized before further processing, thereby leaving the metering terminal vulnerable to code injection attacks.
Metering node infected by malware	Outdated antivirus protection on metering node	The metering node is connected to the Internet in order to communicate with the central system and is therefore susceptible to malware. However, the virus protection on the metering node is rarely updated.
Illegitimate control data sent to the choke components from the central system	Four-eyes principle not implemented, no logging of actions of individual central system operators	The operating procedures and technical implementation of the central system do not enforce approval of control data by a second authorized person. An operator is therefore able to send control data that are not legitimate. Moreover, there is no logging of the actions of individual operators.

Table 16: Vulnerabilities with respect to malicious threats.

The important part, however, is to relate the vulnerability to one of the threats we introduced earlier. With the above in mind: Risk = asset + threat + vulnerability. If there is no link between these 3 then there is no associated risk.

8.2.1.4 Incident Identification

Let us now link the threat to the asset involved. The incident is what links the exploit of a vulnerability to that specific asset. We then list all the threats, all the assets and make the Cartesian product to find all the possible incidents that make sense.

Threat	**Incident**	**Asset**
DDoS attack on the central system	Data from metering nodes cannot be received by the central system due to DDoS attack	Availability of meter data
Tampering with all or most control data in transit from the central system to the choke component	False control data received by all or most choke components	Provisioning of power to electricity customers
Tampering with data in transit from the metering terminal to the central system	False meter data for a limited number of electricity customers received by the central system	Integrity of meter data
Malware to manipulate meter data is installed on the metering terminal through connection to the external meter	Same as the row above	Same as the row above
Metering node infected by malware	Malware compromises meter data	Integrity of meter data
Metering node infected by malware	Malware disrupts transmission of meter data	Availability of meter data
Metering node infected by malware	Malware disrupts the choke functionality	Provisioning of power to electricity customers
Tampering with control data in transit from the central system to the choke components for selected electricity customers	False control data received by the choke components for selected electricity customers	Provisioning of power to electricity customers
Illegitimate control data sent to the choke components from the central system	Power supply to electricity customers is switched off without legitimate reason	Provisioning of power to electricity customers

Table 17: Incidents caused by malicious threats.

8.2.2 Identification of Non-malicious Cyber-risk

We start from the incident to get to the threat sources.

8.2.2.1 Incident Identification

Asset	Incident	Description
Provisioning of power to electricity customers. Availability of meter data	Communication between the central system and the metering terminal is lost	Provisioning of power to the electricity customer depends on control data being sent from the central system to the metering terminal. Availability of meter data depends on such data being sent in the opposite direction
Integrity of meter data	Software bug on the metering terminal compromises meter data	Metering terminals run software to register meter data and transmit these to the central system. Software bugs on metering terminals may therefore compromise meter data
Availability of meter data	Software bug on the metering terminal disrupts transmission of meter data	Similarly to the above case, software bugs on metering terminals may disrupt transmission of meter data to the central system
Provisioning of power to electricity customers	Software bug on the metering terminal disrupts the choke functionality	Control signals to the choke component from the central system go via the metering terminal. Software bugs on metering terminals may therefore disrupt the choke functionality by not forwarding correct control signals
Provisioning of power to electricity customers	Mistakes during maintenance of the central system disrupt transmission of control data to the choke component	Maintenance mistakes such as misconfiguration of communication parameters may prevent or disrupt transmission of control data
Availability of meter data	Mistakes during maintenance of the central system prevent reception of data from metering nodes	Maintenance mistakes such as misconfiguration of communication parameters may prevent metering node data from being received
Provisioning of power to electricity customers. Availability of meter data	The metering terminal goes down due to damage from lightning	Lightning may result in physical damage to the metering terminal which prevents it from functioning

Table 18: Incidents caused by non-malicious threats.

8.3 Risk Analysis

Let's now proceed with the actual quantification. We calculate the value of each risk and work out the risk matrix. Recall that for what was said at the beginning:
Risk Estimation = Likelihood Estimation + Consequences Estimation. For each risk, we must evaluate consequences and likelihood. Therefore, the analysis of the penultimate and last table is fundamental. To proceed in this sense, it is essential to acquire information. We will use the OWASP risk rating methodology. We then answer 4 questions to quantify the value of the risk and impact:

1. **How likely threats to materialize?** Answered in terms of the defined frequency scale.
2. **How severe are the vulnerabilities?** Answered in terms of high/medium/low.
3. **How likely are the incidents to occur?** Answered in terms of the defined frequency scale.
4. **What is the impact of the incidents on assets?** Answered in terms of the defined consequence scales.

8.3.1 Risk Analysis Process - How likely threats to materialize?

8.3.1.1 Malicious Threat Analysis

We choose to follow an approach inspired by the OWASP risk-rating method [10]. The factors are rated on a scale from 0 to 9 (Table 5).
It is not a rigid assignment of values: we can also use values outside those proposed. Let's now analyse each row of the Table 15, focusing only on the first line we would get:

- **Threat**: DDoS attack on the central system
- **Threat Source**: Script kiddie

Factors	Score [1,9]	Rationale
Skill Level	3	She/he is relatively unskilled and unable to perform complicated attacks
Motive	1	Motive generally weak
Opportunity	7	She/he has enough resources and opportunities to conduct the attack (low cost)
Size	7	Script kiddies can reside anywhere in the world
AVG	4.5	

The value of the opportunity is high in the sense that the cost of a DDoS attack is very low and therefore we can do it safely. The size value is also high in the sense that the number of script kiddies in the world is very high (the higher the number of potential attackers of that type, the higher that number will be). The numbers must however take into account the previously created tables. Let's now see the same threat, moved from a different source:

- **Threat**: DDoS attack on the central system
- **Threat Source**: Cyber-terrorist

Factors	Score [1,9]	Rationale
Skill Level	7	May command significant resources and skill, in some cases even being supported by nation states. Able to perform long-term planning, preparation, and carrying out of attacks.
Motive	8	
Opportunity	7	
Size	3	Cyber-terrorist are far less than script kiddie
AVG	6.25	

Based on these results (4,5 and 6,25), considering the worse case, and using our own likelihood scale, we estimate the likelihood of this threat to be "likely":

Likelihood value	Description	
Rare	Less than once per ten years	0 - 1,8
Unlikely	Less than once per two years	1,8 - 3,6
Possible	Less than twice per year	3,6 - 5,4
Likely	Two to five times per year	5,4 - 7,2
Certain	Five times or more per year	7,2 - 9

So DDoS attacks from script kiddies or cyber-terrorists both have a likelihood equal to "likely". If the calculated data are not consistent with the reality of the facts, we can change the value by raising or lowering it if necessary. Eventually, we will have a likelihood table like this:

Threat	Likelihood
DDoS attack on the central system	Likely
Tampering with all or most control data in transit from the central system to the choke component	Possible
Tampering with data in transit from the metering terminal to the central system	Possible
Malware to manipulate meter data is installed on the metering terminal through connection to the external meter	Possible
Metering node infected by malware	Rare
Tampering with control data in transit from the central system to the choke components for selected electricity customers	Unlikely
Illegitimate control data sent to the choke components form the central system	Unlikely

8.3.1.2 Non-Malicious Threat Analysis

Incident	Threat	Entry point
Communication between the central system and the metering terminal is lost	Internet connection to the metering terminal goes down	Internet connection to the metering terminal
Software bug on the metering terminal compromises meter data	Buggy software distributed on metering terminals	Metering terminal
Software bug on the metering terminal disrupts transmission of meter data	Same as the row above	Metering terminal
Software bug on the metering terminal disrupts the choke functionality	Same as the row above	Metering terminal
Mistakes during maintenance of the central system disrupt transmission of control data to the choke component	Mistakes during update/maintenance	Central system
Mistakes during maintenance of the central system prevent reception of data from metering nodes	Same as the row above	Central system
The metering terminal goes down due to damage from lightning	Electricity customer home/building is struck by lightning	Metering terminal

Table 19: Non-malicious threats.

For each identified threat, start by considering the threat source (Table 20).

Threat	Threat source	Description
Internet connection to the metering terminal goes down	Internet connection to the metering terminal	Problems with the connection may initiate threats to the communication between the metering terminal and central system
Buggy software distributed on metering terminals	Software bug	Any kind of software error or malfunction that arises due to mistakes rather than malicious intent
Mistakes during update/-maintenance of the central system	Maintenance personnel	Persons responsible for maintaining the computer systems and infrastructure for the distribution system operator. They do not seek to cause harm, but may still do so by mistake, neglect, or lack of proper training. Notice that a maintenance person with malicious intent is considered to be an insider with respect to this risk assessment
Electricity customer home-/building is struck by lightning	Lightning	Strokes of lightning which may have potential for causing damage in computerized systems and network infrastructure

Table 20: Non-malicious threat sources.

To estimate the Likelihood, it is possible to consider information such as event logs, expert judgments, interviews or questionnaires, and available statistics about the typical likelihood of similar threats in enterprises and other organizations (Table 21).

Threat	Likelihood	Estimate basis/comments
Internet connection to the metering terminal goes down	Certain	This includes cases where individual electricity customer's homes lose Internet connection, which according to general statistics happens very often
Buggy software distributed on metering terminals	Possible	TThis estimate is based on patching logs for various software products developed by the provider of metering terminal software during the last four years
Mistakes during update/maintenance of the central system	Certain	This estimate is based on event logs and statements from the head on the management team
Electricity customer home-/building is struck by lightning	Certain	This estimate is based on statistics for the geographical area where the electricity customers are located

Table 21: Non-malicious threat analysis.

8.3.2 Risk Analysis Process - How severe are the vulnerabilities?

Also in this case we refer to the table elaborated by OWASP to convert the value found into the scale imposed by the answer to be given (HIGH, MEDIUM, LOW):

Severity	Score
0 – 3	LOW
3 – 6	MEDIUM
6 – 9	HIGH

First of all, we need to find a number for 4 attributes of each vulnerability, according to OWASP (Table 6).

8.3.2.1 Malicious Threat Vulnerabilities

As a result of the analysis done in step 2 Risk Identification, we identified 5 different vulnerabilities associated to the malicious threats:

1. Inadequate attack detection and response on central system.
2. Weak encryption and integrity check.
3. Unprotected local network, no sanitation of input data from the external meter.
4. Outdated antivirus protection on metering node.
5. Four-eyes principle not implemented, no logging of actions of individual central system operators.

And following the directives of OWASP, for the first, for example, we find:

Factors	Score [1,9]	Rationale
ease of discovery	7	Checking whether systems are vulnerable to DDoS attacks is often straightforward
ease of exploit	5	After conducting tests, we verified that this vulnerability can be exploited
awareness	6	Knowledge of the existence of such vulnerabilities is widespread
intrusion detection	7	After conducting tests we verified that intrusions are usually not detected when they happen
AVG	6.25	

Iterating over all the Vulnerabilities identified we get the following estimation:

Vulnerability	**Severity**
Inadequate attack detection and response on central system	High
Weak encryption and integrity check	Medium
Unprotected local network, no sanitation of input data from the external meter	Medium
Outdated antivirus protection on metering node	High
Four-eyes principle not implemented, no logging of actions of individual central system operators	High

8.3.2.2 Non-malicious Threat Vulnerabilities

For the non-malicious threats there is no intent to discover and exploit vulnerabilities. We try to understand the extent to which there is a lack of barriers that could prevent threats from leading to incidents. As a result of the analysis done in step 2 Risk Identification, we identified 4 different vulnerabilities associated to non-malicious threats:

1. Single communication channel between central system and metering terminal
2. Poor Testing
3. Poor training and heavy workload
4. Inadequate overvoltage protection

In this case the assessment is done analysing the environment and making consideration over the processes in place:

Vulnerability	**Severity**	**Explanation**
Single communication channel between central system and metering terminal	High	The Internet connection is the only communication channel to the central system for many electricity customers
Poor testing	Medium	Inspection of maintenance logs revealed a number of instances where bugs have been discovered in the metering terminal software. Previous experience indicates that the testing routines of the external software provider are unsatisfactory, and the central system operator does not test software updates for metering terminals before deployment
Poor training and heavy workload	Medium	Interviews indicate that security awareness is not high. Key persons have too much to do. Routines for reviewing and testing updates to the central system before deployment are strong
Inadequate overvoltage	High	The computing hardware of metering terminals is not robust with respect to transient overvoltage

8.3.3 Risk Analysis Process - How likely are the incidents to occur?

In order to estimate the likelihood of incidents, we consider the analysis of threats that lead to the incidents and the vulnerabilities that the threats exploit.

8.3.3.1 Likelihood of Incidents caused by malicious threats

At the end of the Analysis done in step 2 Risk Identification, we identified incidents showed in Table 17. Analyzing the case of the DDoS we have:

Incident	Data from metering nodes cannot be received by the central system due to DDoS attack	Likelihood: Likely
Threat	DDoS attack on the central system	Likelihood: Likely
Vulnerability	Inadequate attack detection and response on central system	Severity: High

Let's do some considerations:

- Event logs show only two such incidents for the last three years (which corresponds to Possible).
- However, there is an increasing trend of this type of incidents.
- Although the number of DDoS attacks that succeed will likely be lower than the number of attempts, we still estimate that the frequency for the incident also lies within the interval of Likely on our scale.

8.3.3.2 Likelihood of Incidents caused by non-malicious threats

At the end of the Analysis done in step 2 Risk Identification, we identified incidents showed in Table 19.

Incident	Mistakes during maintenance of the central system disrupt control signals to the choke component	Likelihood: Unlikely
Threat	Mistakes during update/maintenance of the central system	Likelihood: Certain
Vulnerability	Poor training and heavy workload	Severity: Medium

Incident	Mistakes during maintenance of the central system prevent reception of data from metering	Likelihood: Possible
Threat	Mistakes during update/maintenance of the central system	Likelihood: Certain
Vulnerability	Poor training and heavy workload	Severity: Medium

Let's do some considerations:

- At first glance the two incidents seems to occur with the same frequency.
- However, we found before that there are routines in place for reviewing and testing the system before changes are launched.
- Considering that provisioning of power to the electricity customer is more critical than the continuous reading of meter data, the routines are stronger with respect to updates and changes that may affect control data.
- This observation, combined with the data logs, leads us to the likelihood "Unlikely" regarding control data to the choke component, and the likelihood "Possible" regarding the reception of meter data.

8.3.4 Risk Analysis Process - What is the impact of the incidents on assets?

Let's go back to the Table 17 and take the Impact Scale Table 11 for the first asset. Also considering that, historically, no DDoS attack has ever caused a loss of availability higher than one day and that the number of users whose attacks can increase together with the new number of customers, we assign a "Moderate" estimate and proceed like this for every incident. Until we get the following Table 22:

No.	Incident	Asset	Likelihood	Consequence
1	Data from metering nodes cannot be received by the central system due to DDoS attack	Availability of meter data	Likely	Moderate
2	False control data received by all or most choke components	Provisioning of power to electricity customers	Unlikely	Critical
3	False meter data for a limited number of electricity customers received by the central system	Integrity of meter data	Likely	Minor
4	Malware compromises meter data	Integrity of meter data	Rare	Moderate
5	Malware disrupts transmission of meter data	Availability of meter data	Rare	Moderate
6	Malware disrupts the choke functionality	Provisioning of power in electricity customers	Rare	Major
7	False control data received by the choke components for selected electricity customers	Provisioning of power to electricity customers	Rare	Insignificant
8	Power supply to electricity customers is switched off without legitimate reason	Provisioning of power to electricity customers	Unlikely	Moderate
9	Communication between the central system and the metering terminal is lost	Provisioning of power to electricity customers	Certain	Minor
10	Same as the row above	Availability of meter data	Certain	Insignificant
11	Software bug on the metering terminal compromises meter data	Integrity of meter data	Unlikely	Moderate
12	Software bug on the metering terminal disrupts transmission of meter data	Availability of meter data	Unlikely	Moderate
13	Software bug on the metering terminal disrupts the choke functionality	Provisioning of power to electricity customers	Rare	Major
14	Mistakes during maintenance of the central system disrupt transmission of control data to the choke component	Provisioning of power to electricity customers	Unlikely	Moderate
15	Mistakes during maintenance of the central system prevent reception of data from metering nodes	Availability of meter data	Possible	Minor
16	The metering terminal goes down due to damage from lightning	Provisioning of power to electricity customers	Likely	Insignificant
17	Same as the row above	Availability of meter data	Likely	Insignificant

Table 22: Likelihood and consequence for incidents caused by malicious and non-malicious threats.

8.4 Risk Evaluation

There are 4 main steps (not too much different from the general case):

1. Consolidation of risk analysis results
2. Evaluation of risk level
3. Risk aggregation
4. Risk grouping

8.4.1 Consolidation of Risk Analysis Results

The goal of this activity is to make sure that the correct risk level is assigned to each risk. The central question is not whether each likelihood and consequence estimate is correct, but rather whether the resulting risk level is correct. Example:

- Malware compromises meter data: we have said that the consequences are moderate and the likelihood is rare. We have therefore assigned a low-level risk (green area). We can make some considerations: if we were wrong to determine the likelihood and this should be equal to unlikely, we could still leave the risk in the low level, so the uncertainty about likelihood would not be so serious. However, if the uncertainty were so much higher and we could cover more than two levels, then this would move the total risk level.
- Mistakes during maintenance of central system prevent reception of data from metering nodes. If we were wrong to determine the consequences and this should be moderate rather than minor, we would have to move the risk level from low to medium we must therefore minimize uncertainty.

We also make sure to check whether there are any risks that are both malicious and non-malicious. This is typically the case if malicious and non-malicious threats can result in the same incident. In our case, this would mean that the same incident occurs in both malicious and non-malicious table. In such cases we need to check that the likelihood and consequence estimates are consistent, and that both the malicious and the non-malicious causes have been considered when estimating the likelihood. This can be easy to overlook since we are dealing with the malicious and non-malicious risks separately during much of the risk assessment.

8.4.2 Evaluation of Risk Level

The risk level of each risk is determined by its likelihood and consequence according to the risk matrix. In our case, risk evaluation is performed simply by plotting each risk in the risk matrix Table 23:

		Likelihood				
		Rare	Unlikely	Possible	Likely	Certain
Consequence	Critical		2			
	Major	6				
	Moderate	4,5	8		1	
	Minor				3	
	Insignificant	7				

Table 23: Risk Evaluation Criteria malicious threats.

Risk 1 and risk 2 are high-level risks, risk 3 is medium level and from 4 to 8 are low-level risks. Many of the identified risks are therefore not as important as being in the green zone. However, if we made a few more considerations regarding risk 6 and made some mistakes due to uncertainty, we would move its position to **medium** level or even **high** level. The same goes for risk number 8. Instead, for risk number 3, the situation is even more delicate: it is sufficient for uncertainty to affect only one of likelihood and consequence that this risk moves to a high level. We need further

investigation. Let's examine risk 6, taking into account that malware is expanding widely, likelihood is not necessarily rare and therefore we redefine it as unlikely, the risk thus moves into the yellow zone. The same considerations can be made for non-malicious threats, so we will have:

		Likelihood				
		Rare	Unlikely	Possible	Likely	Certain
Consequence	Critical					
	Major	13				
	Moderate		11,12,14			
	Minor			15		9
	Insignificant				16,17	10

Table 24: Risk Evaluation Criteria non-malicious threats.

In this case, as regards the non-malicious risks, we see that only one, risk 9 has a high level, risk 10 has a medium level and all the others have a low risk. The highest risk is linked to the failure of the connections between the central system and metering terminals, in particular, it has such a high level because it has certain likelihood and minor consequences. In this case, we cannot consider the risk stable because we have been too conservative with regards to likelihood and therefore we can lower it from certain to likely, bringing the risk from **high** to **medium**. As for the risk 10, of medium level, although this is of the same type as the 9, we leave it like this, because as likely as the failure between a single user and the provider may be, it will not create many problems.

8.4.3 Risk Aggregation

During the evaluation we need to take into account that some risks may "pull in the same direction" to the degree that they should actually be evaluated as a single risk. There are basically two cases where this may hold:

- The same incident damages two different assets: where the incident is listed only once in the table. The likelihood remains the same, while the consequences we combine them.
- Two incidents damage the same asset: where the consequences don't change. The likelihood is summarized together to consider the coexistence of these scenarios concurrently.

We use the incident Table 22 and we see that, in the non-malicious risks section, we have for 9 and 10 that is the same incident that damages two different assets (unique in this exercise). However, here we are talking about two non-malicious incidents, and there is no point in aggregating them as they are both random. Let's now analyze risk 4 (malicious) and risk 11 (non-malicious). Here two incidents damage the same asset (meter data integrity):

Risk	**Likelihood**	**Consequence**
(4) Malware compromises meter data	Rare	Moderate
(11) Software bug on the metering terminal compromises meter data	Unlikely	Moderate
(4 + 11) Software on the metering node compromises meter data	Possible	Moderate

Both consequences are moderate and in this case, although these two events act by affecting the same asset, we leave the same consequence. What will change is the overall likelihood: one of the two is rare and the other is unlikely, but if we put them together, and configure the risk with the broader wording, such as "Software on the metering node compromises meter data", the likelihood of this becomes as high as possible. Similar considerations can also be made for the risk pairs 5,12 and 6,13. For the rest of the risks, we have decided to leave them isolated. In the end, we will get the new risk matrix with the union of the two types of risk (malicious and non-malicious):

		Likelihood				
		Rare	Unlikely	Possible	Likely	Certain
Consequence	Critical		2			
	Major	6,13	(6+13)			
	Moderate		8,11,12,14	(4+11),(5+12)	1	
	Minor			15	3	9
	Insignificant				16,17	10

Table 25: Risk Evaluation Criteria for malicious and non-malicious threats.

As can be seen, the Table 25 shows both the aggregate risk and the individual risks that make it up. This is to keep the vision as broad as possible, by doing so we will be facilitated in the Risk Treatment. In fact, for budgetary reasons we have available, we will have to decide whether to treat the aggregate risk in its entirety (expensive), or whether to treat only one of the two underlying risks to lower the level of the overall one (generally cheaper).

8.4.4 Risk grouping

Here the grouping takes into consideration the risks that can benefit from the same treatment. We will take the table used above and see if there are pairs of risks that we can group since they have the same treatment in common to be mitigated. For example, risks 14 and 15 benefits from the same type of treatment (Table 22). To carry out the process in a structured way, we take up the matrix containing also the aggregate risks. In particular, when we have to prioritize the risks to be treated, we will give priority to the treatment of the risk 14 because in this way we will also lower the level of risk 15. In this case, their treatment is not the absolute priority, because there are risks with level higher.

8.5 Risk Treatment

The final step of the cyber-risk assessment starts with the identification of treatments for selected risks. We then assess the effect of the treatments and consider whether the residual risk is acceptable. If it is, the documentation is finalized and the process terminates, otherwise, we need to go back and do another iteration of the treatment identification.

8.5.1 Treatment Identification for malicious risks

Ideally, we would like to find treatments for all identified risks. However, as we always have limited time and resources, we need to focus on the most important ones. We then start by selecting the risks based on the results of the risk assessment and for each risk, however, we identify the treatment and create a table of this type:

Element	**Description**
Risk n.	1
Incident	Data from metering nodes cannot be received by the central system due to DDoS attack
Asset	Availability of meter data
Threat source	Script kiddie; Cyber-terrorist
Threat	DDoS attack on the central system
Attack point	Internet connection to the central system
Vulnerability	Inadequate attack detection and response on central system
Treatment	**Implement state-of-the-art DDoS attack detection and response mechanism on central system**

8.5.2 Risk Acceptance

Now, we have concrete mitigation and we have to decide whether we can accept it or not. To decide we can rely on two approaches:

- **Quantity**: We should try to estimate the costs numerically and accurately (Cost of mitigation, cost of the actual accident, etc.).
- **Qualitative**: A higher level estimate in which we estimate very briefly the costs of mitigation and the accident and then decide what to do.

In this case, we opt for a qualitative approach because we have said that the assessment is of a high level for management, we have not included many details, we have not included figures, numbers, etc.

8.5.3 Cost-Benefit Analysis

Considering the Table 26 we can do some consideration:

Risk	Data from metering nodes cannot be received by the central system due to DDoS attack
Risk Level	High
Treatment	Implement state-of-the-art DDoS attack detection and response mechanism on central system

Table 26: DDoS attack risk and treatment.

- From the point of view of likelihood, we lower the value from **likely** to **possible** as the script kiddies, being compliant, and seeing that the DDoS attack does not work will stop executing it, unlike the cyber-terrorists, who will continue to attack despite everything.
- From the point of view of the consequences, we lower the value from **moderate** to **minor** as with this mechanism we greatly reduce the attack range of several users.

In fact, with this mitigation, we lower the risk from **high** to **low**. However, the cost of the treatment is high. By repeating the cost/benefit analysis for all risks, we get Table 27:

Treatment	**Risk**	**Effect**	**Cost**
Implement state-of-the-art DDoS attack detection and response mechanism on central system	1	High to Low	High
Stronger integrity checking of received meter data on central system	4	Low to Low	High
	11	Low to Low	
	4+11	Medium to Low	
Hire more staff	14,15	Low to Low	High
Develop executable scripts for routine maintenance tasks	14,15	Low to Low	Low

Table 27: Effect of treatments.

It is worthwhile to apply risk mitigation 4 + 11 rather than 4 and 11 separately. However, surely we will have to fix risk 1 and if we have enough budget also 14 and 15, the one with the lowest possible cost.

9 Introduction to Threat Modelling

Threat modelling is a structured approach to identify, quantify, and address threats. It allows system security staff to communicate the potential damage of security flaws and prioritize remediation efforts. The support it provides to risk management is obvious, but it is not the only one, for example, it helps to strengthen the protection of the system on which it is conducted, improves preparation, in particular, we accustom our company to certain types of situations that can result from an accident, and ultimately improves awareness.

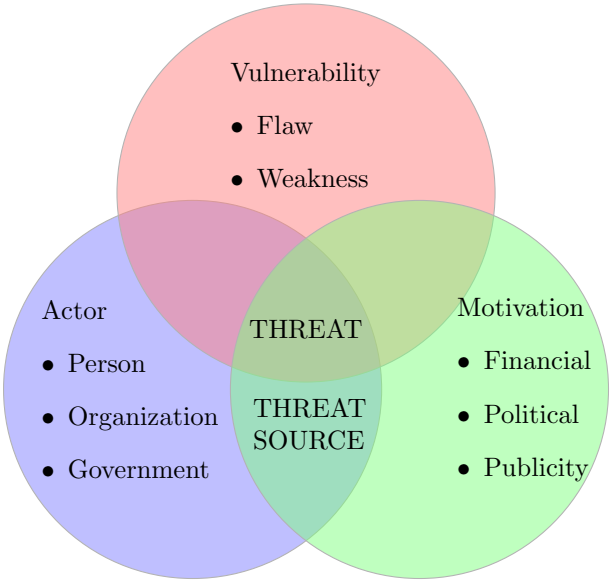


Figure 25: The threat is the intersection between vulnerability, actor and motivation. The intersection between actors and motivation is the threat source.

The threat can be considered as an input to the risk calculation. In fact, if we move from the two-factor risk calculation to the three-factor one, we get:

$$Risk = Threat \cdot Probability \cdot Impact$$

9.1 Learning to Threat Model

We begin threat modelling by focusing on four key questions:

1. What are you building?
2. What can go wrong?
3. What should you do about those things that can go wrong?
4. Did you do a decent job of analysis?

9.1.1 What are you building?

What is the system we want to create?
What is the system that we already have and that we want to evaluate?
We list all the functionalities, how these are translated in terms of practical processes and implemented (internally and externally) with brainstorming, diagrams etc. Obviously, the most important part here is to become aware of the things that the organization directly controls, develops within a known environment and things that are outside this limit of trust.

9.1.2 What can go wrong?

A simple way to answer is using the **STRIDE** Mnemonic to Find Threats. The name takes it from the acronym of 6 threats, the most relevant:

- **Spoofing**: pretending to be something or someone you're not.

- **Tampering**: modifying something you're not supposed to modify.
- **Repudiation**: claiming you did not do something (regardless of whether you did or not).
- **Information Disclosure**: exposing information to people who are not authorized to see it.
- **Denial of Service**: attacks designed to harm system availability.
- **Elevation of Privilege**: when a program or user is technically able to do things that they're not supposed to do.

Once the 6 types of threats have been realized, let's see which of them can affect external entities, and therefore not considered trusted, but also the interactions between external functions and those of the trust boundaries. Whenever we notice a vulnerability, we take note of it, even if it is not part of what we were initially considering. Among all those found, we will then focus only on those that are objectively possible, because otherwise the list of threats becomes overestimated.

9.1.3 What should you do about those things that can go wrong?

We identify effective and efficient countermeasures.

9.1.4 Did you do a decent job of analysis?

We check the model we have come up with and simply respond.

The answers to these last two questions will be more detailed in later chapters.

9.2 Strategies for Threat Modeling

Threat modeling variants:

- Asset-centric
- Attacker-centric
- Software-centric

9.2.1 Asset-centric TM

To be used when we want to focus on the consequences of risks affecting assets, on the business process side. The focus of the analysis is on the set of assets and on everything that can go wrong in the role of the defender, who must identify what is important to the organization, and what can go wrong in the role of the attacker, who must identify the assets it wants to target. Operationally speaking, we will follow these 3 phases:

1. Make a list of your assets and then consider how an attacker could threaten each;
2. connect each item on the list to particular computer systems or sets of systems;
3. draw the systems (showing the assets and other components as well as interconnections) until you can tell a story about them.

We can use this model to apply either an attack set like STRIDE or an attacker-centered brainstorm to understand how those assets could be attacked.

9.2.2 Attacker-centric TM

To be used when we are trying to improve awareness and/or want to provide a remediation plan. Here it is imperative to think like a striker. We start with all the possible attack techniques, we analyze what can be the possible set of attackers, who can use them, and then we try to understand how to apply what has been said in the system. To support this methodology, we can refer to some repositories, such as CAPEC for example. Typically, this technique allows us to identify threats that are related to human vulnerabilities.

9.2.3 Software-centric TM

To be used when we want to check if we are subject to a specific threat. Here we have two types of scenarios, the ideal world and the real one, starting from the ideal one the requirements and implementations are perfectly overlapped (Figure 26).

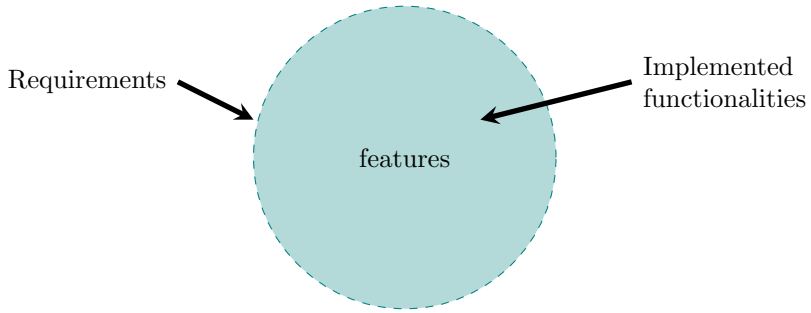


Figure 26: Software development in an ideal world.

In the real world, showed in Figure 27, what happens is that only an intersection respects the implementation and the requirements (features), in fact, we will have two other sections which are:

- bug
- unintended and undocumented functionalities

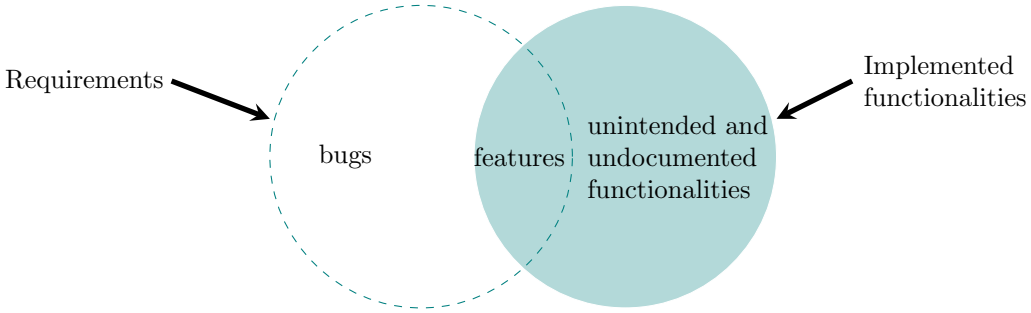


Figure 27: Software development in a real world.

Obviously, bugs and unintended functionalities are vulnerabilities, so we have to find them all and fix them, where possible. Ideally, we need to conduct software-centric analysis during the design period, as this approach is indicated during software design. In reality, however, security is never considered important when designing software. So we need to create a diagram that allows us to see the software and its interactions with external entities:

- Via Flow chart: Although not widely used during threat modelling, this allows us to observe the flow of data well, however it does not allow us to see how the system/software is composed.
- Via Data flow diagram - DFD: It allows us to proceed with a top-down approach. Iterate over the system components. This method focuses on the inputs and thus on the data types exchanged between the components.
- Via Universal Model Language - UML: Represents the system from a different perspective.

9.3 Data Flow Diagrams - DFD

A Data Flow Diagram is a graphical representation of flow of data through an information system DFD can be used to visualize data processing. Inside it we can represent only a few elements:

- what data is input for the system;
- what data is produced as output;
- processing steps;
- data persistence activities.

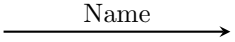
DFDs are useful for threat modelling as they provide an high-level view of how the information system manages assets that need to be protected. Regarding the semantics of DFDs we have 4 elements:

Element	**Appearance**	**Meaning**	**Example**
External entity	web client	People, or code outside your control	Your customer, Microsoft.com
Data flow	HTTP →	Communication between processes, or between processes and data stores	Network connections, HTTP, RPC, LPC
Process	DB control	Any running code	Code written in C, C#, Python, or PHP
Data store	Log DB	Things that store data	Files, databases, the Windows Registry, shared memory segments

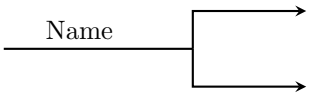
9.3.1 External entity/Terminator

We can put only one, or, if we want to increase the detail we can put several. Usually, they are not part of the system that must be studied, because we do not have direct control over them, i.e. they are not necessarily entities external to the entire organization, but they can simply be outside the department in question. They must receive data from the system and this is usually represented by a data flow.
Rule 1: Never include in a DFD direct data flows from one external entity to another. They are irrelevant to the system being described because they are external.

9.3.2 Data flows

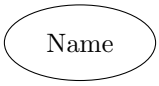


Represent data moving between elements of the system and data movements are always directed.
Rule 2: Only represent data, not material goods and remember that each arrow has a name representing the specific data moved through the flow.
Rule 3: Only include one type of data per arrow.



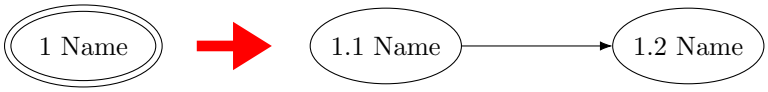
A fork in a data flow means that the same data goes to two destinations. The same data coming from several locations can also be joined.

9.3.3 Processes



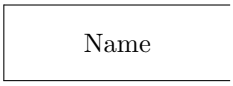
Represent where the business logic is implemented.
Rule 4: Processes must have at least one data flow in and one data flow out. They are named with a verb and an object of the verb. Remember that each process should represent only one function or action.

Processes represented with a double circling line are composite or a high-level aggregate of two more detailed processes. This allows us to create a hierarchical diagram that consists of different levels, usually level 0 is the most general, while it becomes more detailed once we go up with the levels:



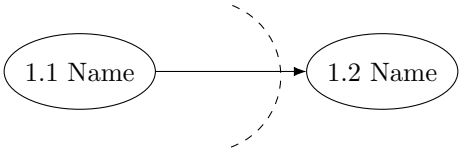
Level 0 DFD Level 1 DFD

9.3.4 Data store



The last element represents the place where the data is stored, be it a repository, cloud, archives, directories, databases, etc. The name chosen for these entities is usually related to the data contained within it. Within the diagram there can be multiple identical instances and can be shared by multiple systems.

9.3.5 Trust boundaries



The dashed line represents the limits in the trustability of the sub-parts of the system.

9.3.6 Mistakes to avoid

In addition to remembering the rules for each element of the system, the following good diagramming practices must be used:

- Do not use direct data flows from one data store to another. There must be a process between the stores.
- Do not use direct data flows from an external entity to a data store. A process is needed between them.
- Do not show direct data flows between external entities.

9.3.7 Building a DFD step-by-step

1. Identify actions and actor with use cases.
2. Build context level DFD:
   We have at the centre of the diagram the system that is under analysis, the main functionality, and around it, we put all the entities that interact with the system, concerning the direction of the data flow.
3. Build Level-0 DFD:
   Here we refine the context level diagram by identifying the main business processes and the data flows that interconnect them. Here we introduce the data stores.
4. Build Level-1 DFD:
   We specify for each of the processes created in the previous step how they collect data and how they process it and we can, if possible, decompose the functionalities of the single process. We must take into account the trade-off between the granularity of the analysis and the number of processes within the system. Avoid abstracting processes too much, but also detailing them too much.
5. Add trust boundaries.

10 STRIDE, Attack trees and Attack libraries

10.1 STRIDE

10.1.1 Spoofing classification

STRIDE falls under the software-centric threat modelling framework and classifies threats as follows:

Threat	Violated Property	Threat Definition	Typical Victims	Example
Spoofing	Authentication	Pretending to be something or someone other than yourself	Processes, external entities	Falsely claiming to be Acme.com, winsock.dll, Barack Obama, a police officer, or the Nigerian Anti-Fraud Group

In spoofing, there is an entity in the system that claims to be someone or something other than itself. Here is an example list of spoof threats:

- Spoofing a process: race conditions, rename file, linking file.
- Spoofing a file.
- Spoofing a Machine: ARP/IP/DNS spoofing, DNS compromise, IP redirection.
- Spoofing a person: sets email display name, takes over a real account.
- Spoofing a role: Declares themselves to be that role.

10.1.2 Tampering classification

Threat	Violated Property	Threat Definition	Typical Victims	Example
Tampering	Integrity	Unauthorized modification of data on disk, on a network, or in memory	Data store, data flows, processes	Changing a spreadsheet, the binary of an important program, or the contents of a database on disk; modifying, adding, or removing packets over a network, either local or far across the Internet, wired or wireless; changing either the data a program is using or the running program itself

In tampering, there is an entity that modifies data on disks, in transit on the network or in memory without authorization. Here is an example list of tamper threats:

- Tampering with a file: Modifies a file on the workstation or server, or modifies redirecting link.
- Tampering with memory: Modifies code or API.
- Tampering with a network: Modifies data flowing over the network.

10.1.3 Repudiation classification

In repudiation, the entity violates non-repudiation property, that is, that which is connected to the capacity to attribute a certain responsibility to each action in the system. Here is an example list of repudiate threats:

- Repudiating an action: Claims to have not clicked/received or have been a fraud victim.
- Attacking the logs: Notices we have no logs, puts attacks in the logs to confuse logs, or a person reading the logs.

Threat	Violated Property	Threat Definition	Typical Victims	Example
Repudiation	Non-Repudiation	• Claiming that you didn't do something, or weren't responsible. • It can be honest or false • Often appears at the business level • The key question for system designers is, what evidence do you have?	Processes	Process or system: "I didn't hit the big red button" or "I didn't order that Ferrari." Note that repudiation is somewhat the odd-threat-out here; it transcends the technical nature of the other threats to the business layer.

10.1.4 Information classification

Threat	Violated Property	Threat Definition	Typical Victims	Example
Information Disclosure	Confidentiality	Providing information to someone not authorized to see it	Processes, data, stores, data flows	The most obvious example is allowing access to files, email or databases, but information disclosure can also involve filenames, packets on a network, or the contents of program memory.

In Information Disclosure, the problem is the confidentiality of the data in the system and consists in providing information to someone who is not authorized to have access to it. Here is an example list of information disclosure threats:

- Information disclosure against a process: Extracts secrets from error messages/error cases.
- Information disclosure against data stores: Takes advantage of inappropriate ACLs or bad database permissions. Finds files protected by obscurity or crypto keys on disk. Gets data from logs/temp-files/swap. Sees interesting information in filenames.
- Information disclosure against a data flow: Reads data on the network. Learns secret by analysing traffic or talking to whom by social network info or watching the DNS.

10.1.5 Denial of Service

Threat	Violated Property	Threat Definition	Typical Victims	Example
Denial of Service	Availability	Absorbing resources needed to provide service	Processes, data, stores, data flows	A program that can be tricked into using up all its memory, a file that fills up the disk, or so many network connections that real traffic can't get through.

In Denial of Service, we affect the availability of the system, in particular, we talk about those threats that undermine the consumption of system resources so that they are no longer able to support the business process. Here is an example list of Denial of Service threats:

- DoS against a process: Absorbs RAM, disk or CPU.
- DoS against data stores: Fills data store up, makes enough requests to slow down the system.
- DoS a data flow: Consumes network resources.

10.1.6 Elevation of Privilege

Threat	Violated Property	Threat Definition	Typical Victims	Example
Elevation of Privilege	Authorization	Allowing someone to do something they're not authorized to do	Processes	Allowing a normal user to execute code as admin; allowing a remote person without any privileges to run code.

In Elevation of Privilege, the authorization property is undermined and thus they allow someone to do something they are not normally authorized to do. Here is an example list of Elevation of Privilege threats:

- Elevation of privilege against a process by corruption the process: Send inputs that the code doesn't handle properly. Gain access to read or write memory inappropriately.
- Elevation through missed authorization checks.
- Elevation through buggy authorization checks.
- Elevation through data tampering: Modifies bits on disk to do thing other than what the authorized user intends.

10.2 STRIDE Variants

10.2.1 STRIDE-per-Element (Microsoft)

With this approach we simply assume that some threats are more common for certain elements of the diagram than for others:

Threat Type	External Entity	Process	Data Flow	Data Store
Spoofing	X	X	X	
Tampering		X	X	X
Repudiation	X	X		X
Information Disclosure		X	X	X
Denial of Service		X	X	X
Elevation of Privilege		X		

10.2.2 STRIDE-per-Iteration

It is an approach to threat enumeration that considers tuples of (origin, destination, interaction) and enumerates threats against them:

#	Element	Interaction	S	T	R	I	D	E
1	Process 1	Process 1 - Data Store 1	X			X		
2	Process 1	Data Store 1 - Process 1	X	X			X	X
	Data Store 1	...			X			
	Data Flow k							
n	External Entity							

Each tuple consists of 3 elements: origin, destination and interaction. For each of them, we associate a threat through STRIDE analysis, making sure to consider all possible threats in the identified category.

10.2.3 DESIST

DESIST stands for:

- **D**ispute (replace Repudiation)
- **E**levation of privilege
- **S**poofing
- **I**nformation disclosure
- **S**ervice denial (replace Denial of Service)
- **T**ampering

10.3 Check your STRIDE-driven model

Once STRIDE or one of its variants is applied, we will have a list of threats found and we will need to verify that what we found is correct and consistent. At this point, we will have to make some considerations: since STRIDE is not a control, but a guideline we will only have to see if our analysis is complete. There are three ways to judge whether we have done finding threats with STRIDE:

- Try to check if we have at least one threat in each category. Since in the system we usually have at least one element for each entity if there is a threat category that is outside, we have probably forgotten to analyze it.
- Often we would like that for each element there is every type of threat, but this is not always possible because maybe we already use countermeasures that cancel that risk.
- To increase comprehensiveness, we use STRIDE-per-element and make sure that for each of the entities there is at least one "X".

10.3.1 Observations

When using STRIDE we are just enumerating the things that might go wrong; the exact mechanisms for how it can go wrong are something we can analyse later. It can be useful to record all possible attacks, even if there is a mitigation in place. STRIDE is not a taxonomy or a classification mechanism, so it is easy to find things that are hard to match with just one STRIDE criteria.

10.4 Attack trees

We are talking about an alternative to STRIDE, but in this case, it is a graphical representation that allows us to observe the security problems affecting a particular functionality of the system. There are several ways to build the attack tree. Typically, attack trees present the origin node as the target of the attack, i.e. the target of the attack, and the more complex the system will be, the more targets the attackers will have and the more trees will be processed with different origin nodes. Each leaf node is represented by a specific attack used to achieve the goal. Some leaf nodes can be refined and become root nodes etc.
There are three ways you can use attack trees to enumerate threats:

1. Use an attack tree someone else created to help you find threats.
2. Create a tree to help you think through threats for a project you're working on.
3. Create trees with the intent that others will use them.

Creating new trees for general use is challenging, **even** for security experts.

10.4.1 Creating New Attack Trees

Let's assume that in the repositories we can't find an attack tree that meets our needs. We will have to create it ourselves and this can be done because maybe our analysis is of a high level and we don't want to go into too much detail about the attacks that can be used, the specific tools, the steps etc. We want a high-level analysis and therefore we can easily create an attack tree. We will have to follow some steps:

1. Decide on a representation
2. Identify the root node
3. Create sub nodes
4. Consider completeness
5. Prune the tree
6. Check the presentation

10.4.2 Attack trees Representations

- **AND** trees where the state of a node depends on all of the nodes below it being true. Represent different steps in achieving a goal.
- **OR** trees where a node is true if any of its sub nodes are true. Represent different ways to achieve the same goal.

10.4.3 Create a Root Node

The root node can be:

- the component that prompts the analysis where the sub nodes should be labelled with what can go wrong for the node.
- an adversary's goal where the sub nodes should be labelled with ways to achieve that goal.
- a problematic state.

There is no best way to create a root node, but it is recommended to create a root node with an attacker goal or high-impact action, use OR trees and draw them into a grid that the eye can track linearly.

10.4.4 Assigning values to leaves

Once we have decided the point of view from which we want to analyze the problem, we assign the values to the leaves, that is, how a potential attack can take place, summarized by a label. Each node will therefore be an attack that according to our context will be possible/impossible, easy/complex, expensive/cheap, intrusive/non-intrusive, legal/illegal, special equipment required or not required and this will facilitate the pruning of the tree. Then we calculate the value of the inner nodes which are a function of their own child nodes up to the root node. Obviously, we can think of putting not Boolean values but numerical values such as: cost of the attack, cost of defense, time to achieve, resources needed to attack, probability of attack success, likelihood that an attacker will try a given attack.

10.4.5 Human-Viewable Representations

Trees can be represented in two ways: as a free-form (human-viewable) model without any technical structure, or as a structured representation with variable types and/or metadata to facilitate programmatic analysis. Attack trees can be drawn graphically or shown in outline form. Graphical representations are a bit more work to create but have more potential to focus attention. In either case, if our nodes are not all related by the same logic (AND/OR), we'll need to decide on a way to represent the relationship and communicate that decision. If our tree is being shown graphically, we'll also want to decide if we use a distinct shape for a terminal node. The labels in a node should be carefully chosen to be rich in information, especially if we're using a graphical tree.

10.5 Attack Libraries

Another way to identify threats within the system under analysis, perhaps in combo with attack trees, is given by attack libraries that allow us to better identify tree leaves. It can be a useful tool for finding threats against the system we are building and we can refer to different attack libraries that differ according to the number of details, the scope and the audience.

Figure 28: The highest level will be the most abstract and the lowest level will be the most detailed.

10.5.1 CAPEC

We can use this very structured set of information for threat modelling in a few ways:

- Review a system under construction with respect to each CAPEC entry or the 15 CAPEC categories.
- Reviewing individual entries is a large task.
- Train people about the breadth of threats.

10.5.2 CAPEC vs STRIDE

- **CAPEC** is a classification of common attacks, more complex to address attacks and richer elicitation technique.
- **STRIDE** is a set of security properties, more easy to fine defences than attack and property-based.

10.5.3 OWASP Top 10

The OWASP Top 10 is a powerful awareness document for web application security. It represents a broad consensus about the most critical security risks to web applications. Project members include a variety of security experts from around the world who have shared their expertise to produce this list. Although the original goal of the OWASP Top 10 project was simply to raise awareness among developers and managers, it has become the de facto application security standard.

10.5.3.1 Which metrics can you extract from OWASP Top 10?

Attackers can potentially use many different paths through our application to do harm to our business or organization. Each of these paths represents a risk that may, or may not, be serious enough to warrant attention. Sometimes these paths are trivial to find and exploit, and sometimes they are extremely difficult.

10.5.3.2 How did OWASP rank threats?

The OWASP Top 10 focuses on identifying the most serious web application security risks for a broad array of organizations. For each of these risks, we provide generic information about likelihood and technical impact using the following simple ratings scheme, which is based on the OWASP Risk Rating Methodology.

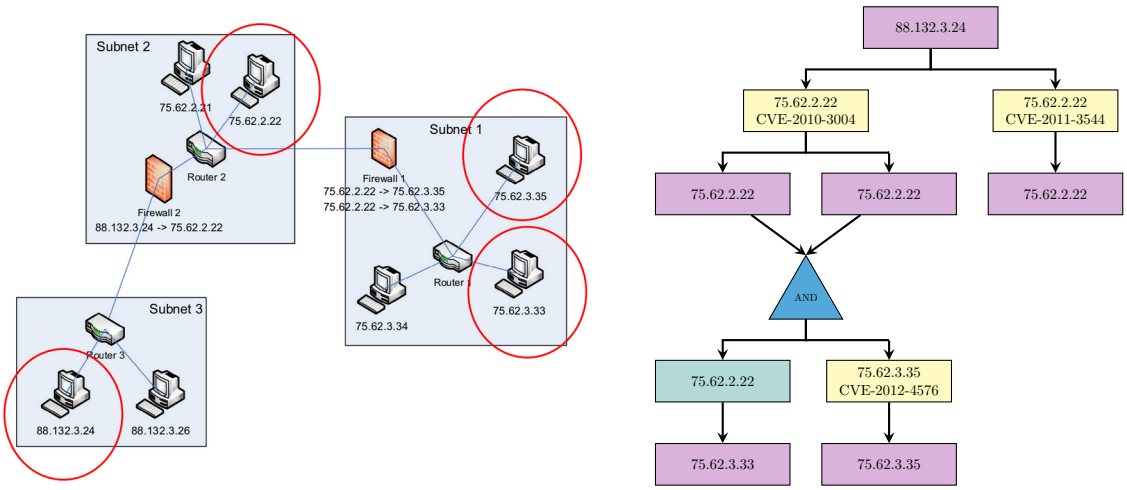
Threat Agents	Exploitability	Weakness Prevalence	Weakness Detectability	Technical Impacts	Business Impacts
Application Specific	Easy: 3	Widespread: 3	Easy: 3	Severe: 3	**Business Specific**
	Average: 2	Common: 2	Average: 2	Moderate: 2	
	Difficult: 1	Uncommon: 1	Difficult: 1	Minor: 1	

11 Attack graph

An attack graph represents possible ways via which a potential attacker can intrude into the target network by exploiting a series of vulnerabilities on various network hosts and gaining certain privileges at each step. Typically an attack graph is composed of:

- Nodes, which represent the host/device in the network and its privileges.
- Edges, which represent a possible vulnerability that we can exploit from a certain node (source) to obtain some kind of privilege on another node (destination). The vulnerability obviously lies on the target node.

The computation of an attack graph requires the computation of the reachability conditions among the network hosts by considering all network protocol layers, modelling attacks and attack paths, and devising an efficient method to compute possibly huge number of attack paths. Let's see an example:



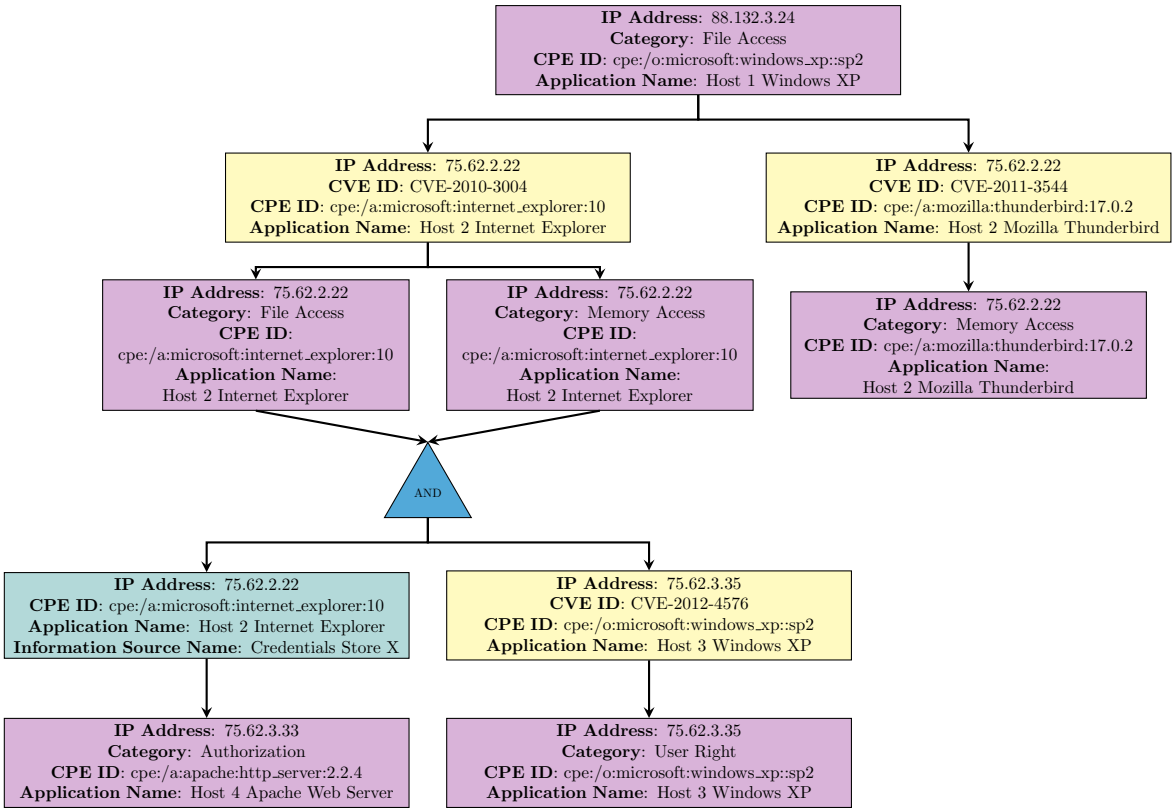
(a) Subnet example (b) Graph example

Figure 29: Attack graph example.

First of all, we choose an arbitrary source to use to elaborate our attack graph and we put it in the purple node. Then we observe that we can only reach the host with IP 75.62.2.22 belonging to subnet 2 and on this, there are two shots (yellow nodes). If we exploited the first one, CVE-2010-3004 we would get: two different levels of privileges. If instead, we exploited the CVE-2011-3544 level we would get only one privilege. With the use of the blue node, which represents an AND, we combine the two types of privileges obtained with the exploit of vulnerability and thanks to this combo we can obtain information (green node) that can lead us to the exploit of another host, in many of the algorithms that we will use this type of node will not be used anyway. For a more detailed graph, we can use the following rules:

- All the nodes, except the junction ones, must contain the following information:
  - The **IP Address** of the node.
  - **CPE ID** which is a standard method for identifying and describing the application for which we are trying to obtain privileges.
  - The **Application Name** which is the user-friendly version of the CPE.
- In the purple nodes (privilege nodes) there is a category that indicates the software system connected to the privilege. For example: file access privileges, memory access privileges and so on.
- In the yellow node (vulnerability node) there is the CVE ID that uniquely identifies the vulnerability we are trying to exploit.

Given these directives, let's see how the above graph becomes:



Based on this information, we can analyze the risk and think about targeted mitigation to block attack paths, but also raise awareness within the organization. It is therefore a very powerful tool.

11.1 Basic problems in attack graph generation

There are 4 main problems in the attack graph generation process:

1. Reachability analysis
2. Attack template determination
3. Attack graph structure determination
4. Attack graph core building mechanism

11.1.1 Reachability analysis

When we begin to analyze the network we know which hosts are in the system and we know how they are physically connected, but there is also other information relating to the routing tables of the routers that are scattered throughout the network, and we must always take them into account when we process the graph because they indicate the reachability conditions between pairs of hosts belonging to different subnets. At this point, we can elaborate a Boolean matrix in which we only define whether two hosts are reachable or simply unreachable, but we can also think of a more complex matrix in which we store more detailed information for reachability based on the services and ports used.

11.1.1.1 Gathering Information for Reachability Matrix computation

The configuration information can include the following:

- the topology of the target network;
- the applications (software or hardware installations) on the network hosts;
- the employed filtering and access control rules;
- the intrusion detection/prevention system configurations;

- trust relations among the network hosts.

The more network configuration information is obtained, the more accurate the attack graphs will be.

11.1.2 Attack template determination

An attack template specifies the conditions required by an attacker to perform a set of specific attacks successfully. It also describes the conditions gained by an attacker, after the corresponding attacks are successfully performed. The attack templates created collectively form the attack model. An attack graph contains the privileges gained on the target network hosts by an attacker and these privileges are related to the possible vulnerability exploits. The relationships between a set of privileges and a vulnerability exploit are determined by using an attack template. Determining what can be a privilege must be done in the attack template design process, and those privileges can be designed based on the type of applications that can be installed on a host computer. As the level of detail of certain privileges increases, the accuracy of the resulting chains of vulnerability exploits in the generated attack graphs increases but also increases the time and space of the requirements of the attack graph core building process.

11.1.3 Attack graph structure determination

Tells us how we define each vulnerability and each information about the devices. We can therefore include various nodes in the graph, as seen, provide various information but in the end, all these must be collected from the environment and must be stored and processed. We must ensure that the processing and storage of information doesn't take too long, otherwise, our efforts become as useless as the information we have collected. For these reasons, when we develop an attack graph, we very often refer only to vulnerabilities and privileges, which for simplicity are understood as those relating to access to the "user" and "root" machine.

11.1.4 Attack graph core building mechanism

In generating both partial and full attack graphs, the initial privileges possessed by the attacker and the target privileges for the attacker are provided as inputs for determining attack paths. For the generation of the full attack graph, every possible attack path from initial privileges to target privileges is found. The process of generating the complete attack graph can be formulated as a general graph traversal problem, as it must find all attack paths. In essence, most of the attack graph generation algorithms proposed in the literature use some form of search algorithm to find matching nodes in the resulting attack graph.

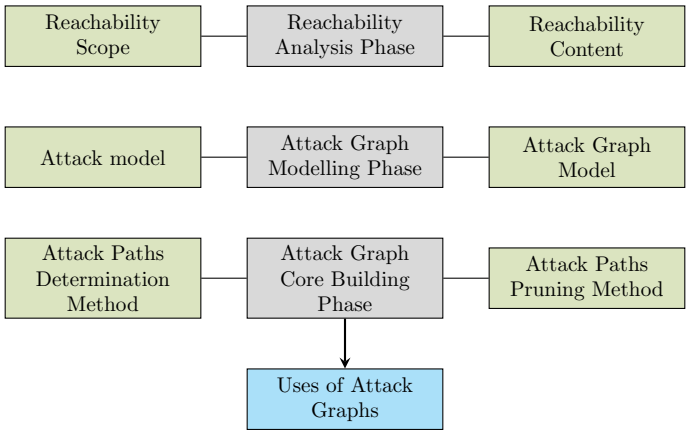
11.1.5 Issues

Low **scalability** due to the fact that we have a lot of information about the system and the more we add, the more complex it becomes. Countermeasures:

- Monotonicity, that is, when we look on the net and acquire a certain level of privilege we will never climb to lower levels, that is an implicit pruning of paths that allow us to climb to lower levels of privileges and then return to higher ones.
- Pruning in which we begin to calculate the attack path and expand it only in the direction in which the metric we have chosen begins to grow.
- Attack paths shorter or develop them to a fixed maximum length.
- Cycle-free attack graph.

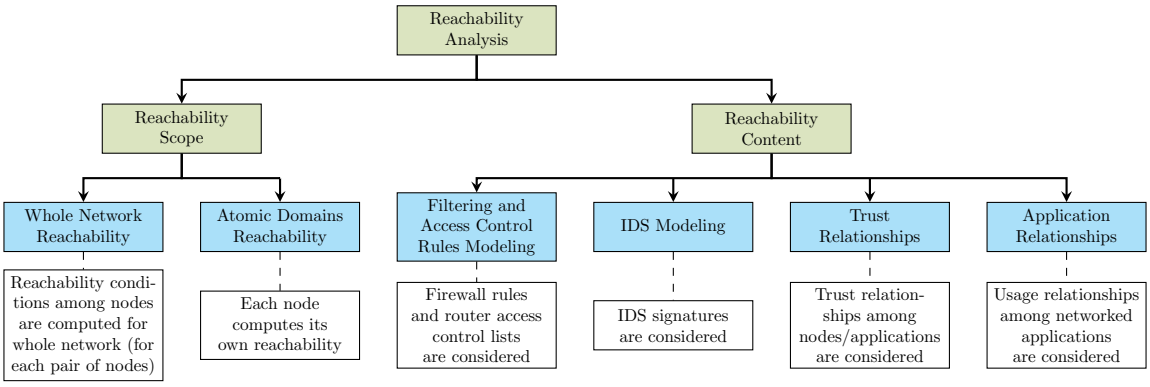
11.2 Attack graph generation process taxonomy

The activities performed during the whole attack graph generation process can be classified into three high-level phases [11]:



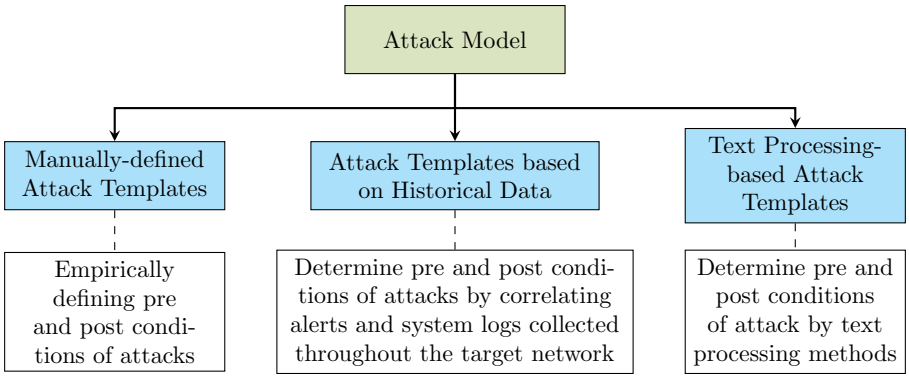
1. **Reachability analysis phase**:
   Two main classification criteria for the reachability information are reachability scope and reachability content.

   - Reachability scope determines the scope of the network hosts among which the reachability conditions are computed before the attack graph core building process.
   - Reachability content determines the network security objects (entities) that are accounted for in the computation of the reachability information.

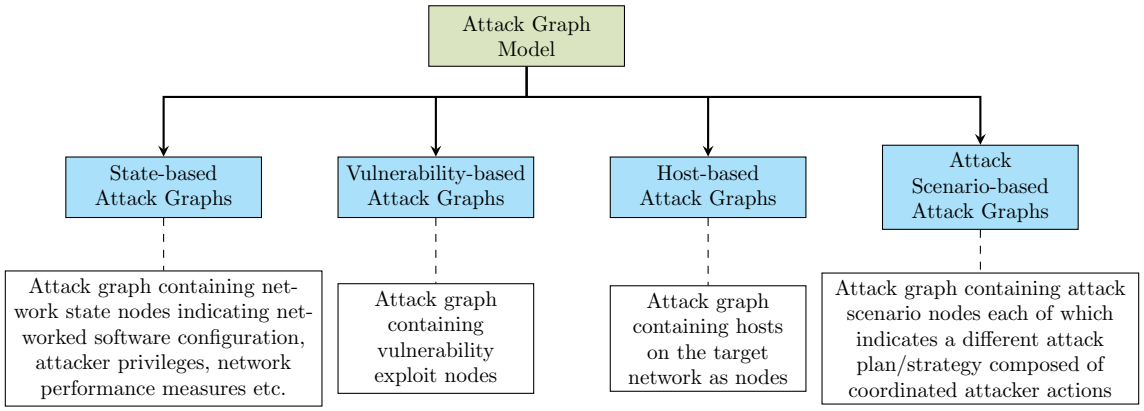


2. **Attack graph modelling phase**:
   In this phase we have two tasks:

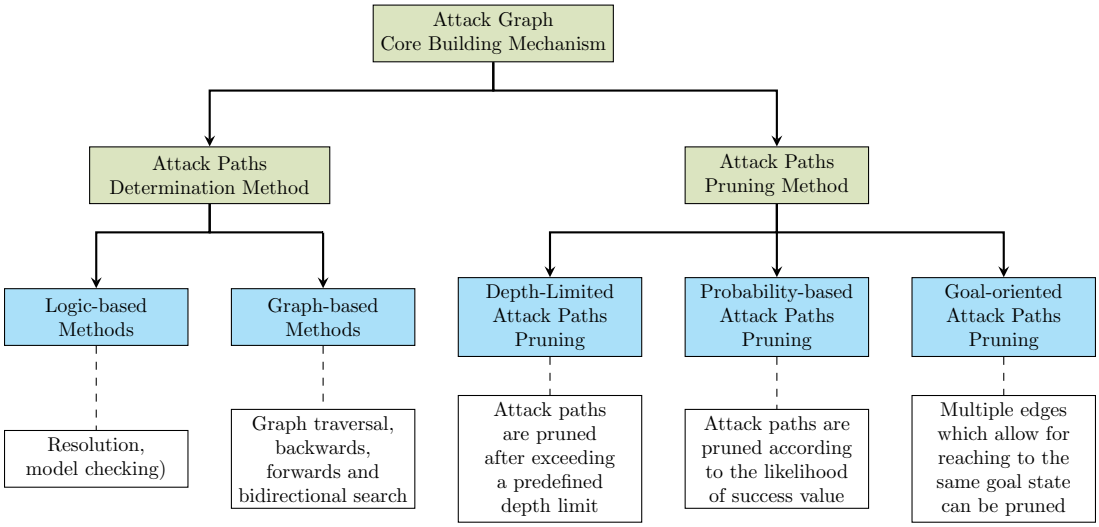
   - **Attack Model**



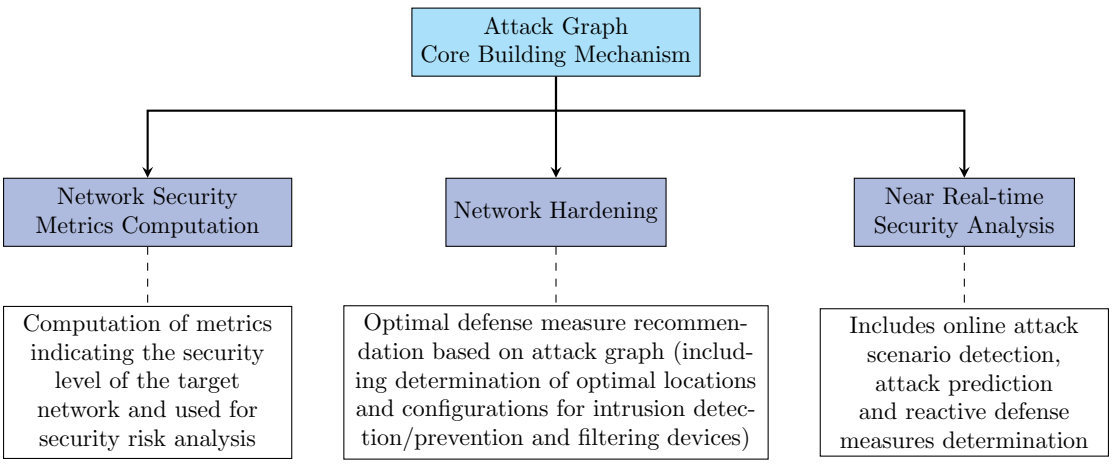
- **Attack Graph Model**



3. **Attack Graph Core Building Phase**



4. **Attack Graph Usage**
   Three possible usage scenarios:



   (a) **Network Security Metrics Computation**:
   We will use it, in our case, to support the risk analysis. We will therefore use an algorithm that creates the attack graph and also measures the risk associated with it. In this case, all we can do is identify the source and target in the attack path system and then calculate

the risk of the specific situation. We do not use a specific threat model because we are not eliciting a particular threat, but in this case, the threats are implicit and are represented by all the attack patterns that we can calculate taking into account vulnerabilities and their possible exploits. So we can calculate the risk based on the threat agent, the different attack paths.

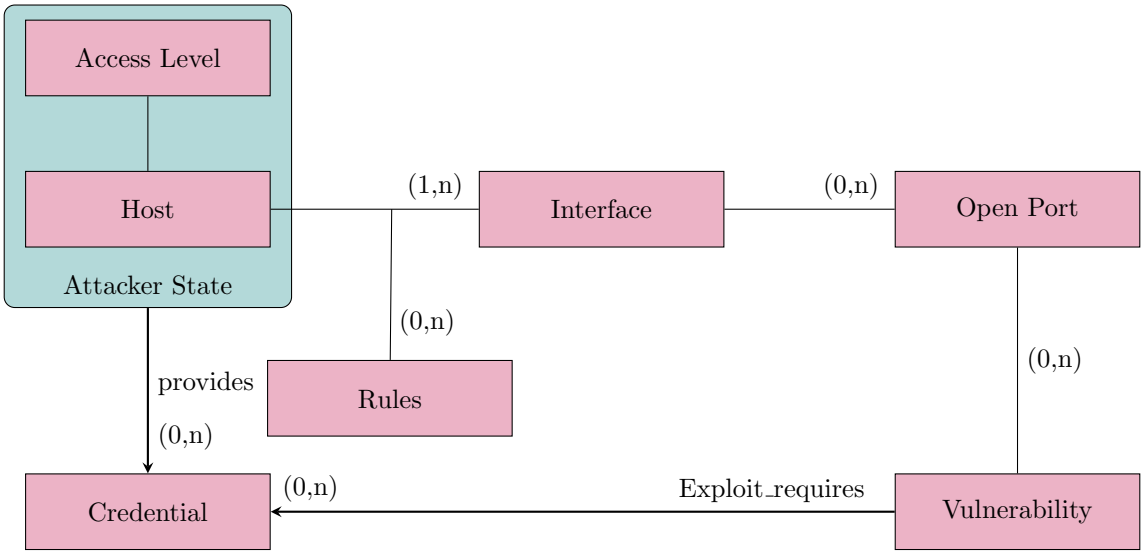
(b) **Network Hardening**:
The idea is to calculate the attack graph to identify the best mitigation method. In this case, the mitigation actions are expressed in terms of vulnerability patches or we can think in terms of reachability.

(c) **Near Real-time Security Analysis**:
Use attack patterns as possible paths that an attacker can follow to perform an exploit and monitor, via IDS alert, whether these attack patterns were followed or not. If we see that an attacker is following a path, then we cut the link of the path to interrupt his offensive.

11.3 NetSPA - Practical attack graph generation for network defense

NetSPA is able to import data from **common sources**, including the Nessus vulnerability scanner, Sidewinder and Checkpoint firewalls, the CVE dictionary, and the NVD vulnerability database. It **automatically computes reachability**, or the ability for a given host to connect to open ports on all hosts in the network. It rapidly generates an attack graph showing how an attacker can maximally compromise the targeted network. The tool can model an attacker able to start from any IP address, maximally exploiting any special "holes" in the perimeter firewalls and builds **multiple-prerequisite graphs** (MP graphs) [12].

11.3.1 NetSPA Data



It all starts with the definition of a host, which is any device on the network. To reach a host, we can associate one or more (1, n) interfaces, which are the ways in which the host can communicate with others and explicitly represents the fact that for each device we can have multiple NICs. The relationship between host and interface is given by the rule entity, which manages the way in which the host communicates through the interfaces. We can obviously have 0 or n rules. In this model, vulnerabilities are not directly associated with the host, but are associated with open ports on specific host interfaces. Obviously, on an interface, a host can have from 0 to n open ports and on an open port from 0 to n vulnerabilities. We define the attacker state as a specific access level on a specific host. Another important aspect that NetSPA highlights are credentials, explicitly defined as what unites the attacker state with vulnerabilities. In particular, the attacker state provides some kind of credential for the system and the credentials are used to perform some exploit of a vulnerability. Vulnerabilities are not only related to software, but it is any weakness in the system that allows the attacker to gain access to the system itself.

11.3.2 NetSPA Multi Prerequisite (MP) Graph

The MP graph uses the following three node types:

- **State nodes** represent an attacker's level of access on a particular host: outbound edges from state nodes point to the prerequisites they are able to provide to an attacker.
- **Prerequisite nodes** represent either a reachability group or a credential: outbound edges from prerequisite nodes point to the vulnerability instances that require the prerequisite for successful exploitation.
- **Vulnerability instance nodes** represent a particular vulnerability on a specific port: outbound edges from vulnerability instance nodes point to the single state that the attacker can reach by exploiting the vulnerability.

11.3.3 NetSPA Graph Construction

The graph is built using a breadth-first technique. No node is explored more than once, and a node only appears on the graph if the attacker can successfully obtain it. Let's see an example of pseudocode for Main Loop:

```
1 BFSQueue starts with the root node(s) representing the attacker's starting STATE(s)
2 while (BFSQueue is nonempty)
3     CurNode = BFSQueue.dequeue()
4     DestSet = all nodes that can be reached from CurNode
5     foreach node DestNode in DestSet
6         add an edge from CurNode to DestNode
7         if DestNode is brand-new,
8         BFSQueue.enqueue(DestNode)
```

The result of the line 4 depends on the type of node under Analysis:

1. CurNode is a state
2. CurNode is a prerequisite that is a reachability group
3. CurNode is a prerequisite that is a credential
4. CurNode is a vulnerability instance

12 Intrusion Detection Systems (IDS)

Given the evolution and dynamic nature of systems, no defence is 100% intrusion-proof. They are in fact always subject to new attack techniques (0-day), unexploited "silent" vulnerabilities, misconfiguration and malicious insiders. Therefore studying the periodic reports is not enough, but we must continuously monitor the system and detect all possible suspicious or abnormal behaviours [13]. More precisely, for NIST, Intrusion Detection is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of intrusions, defined as attempts to compromise the confidentiality, integrity, availability, or to bypass the security mechanisms of a computer or network. This definition provides the environment to be analyzed, the way in which we must analyze it, the way in which we must try to identify strange behaviours and the purpose of these behaviours [14].

12.1 Attack Taxonomy

We need to create a taxonomy of the attacks that can be conducted on the system. When we try to identify an attack we will not only focus on the type but also on other characteristics (Figure 30) [15].

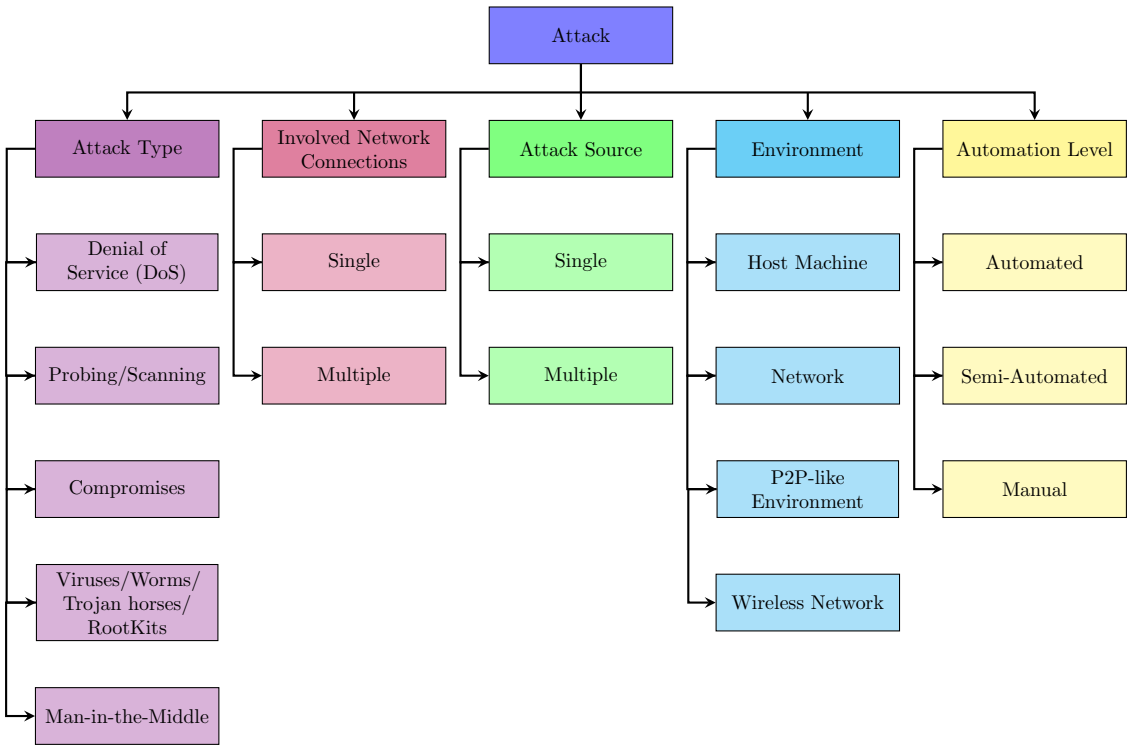


Figure 30: Classification and characteristics of the attacks.

12.1.1 Attack Type

12.1.1.1 Denial of Service (DoS)

- **Goal**: shut down a network, computer, or process; or otherwise deny the use of resources or services to authorized users.
- **Strategies**: Consumption of scarce resources, destruction or alteration of configuration information, physical destruction or alteration of network components.

This category of attack also includes making it impossible for a service to draw on the resources it needs to continue with its task. Prevent access to a service that is alive by changing the ACL. Reduce the level of access privileges so that they do not use part of the services.
Examples of Denial Of Service (DoS) attacks:

- Teardrop, where the attacker exploits the inability to better manage the overlap of fragmented IP packets by sending an enormous amount of fragmented packets.

- SYN flooding, where the attacker sends a series of SYN packets to a server, which responds with SYN/ACK for each of them, then commits its resources waiting for a series of ACK packets that will never arrive.
- Simple Service Discovery Protocol (SSDP) attack, with which the attacker exploits the UPnP (Universal Plug and Play) protocol, thanks to which requests are amplified up to 30 times on the target.

12.1.1.2 Probing/Scanning

- **Goal**: identify valid IP addresses in a domain and collect information about them (e.g. what services they offer, operating system used). This information provides an attacker with the list of potential vulnerabilities that can later be used to perform an attack against selected machines and services.
- **Strategies**: Use specific tool such as IPsweep, Portsweep, nmap.

Countermeasures are very effective in detecting fast and disperse noisy scan (look for IPs that make more than N connections in T seconds), but stealthy scans are more challenging, this type of attack often doesn't have an effective solution.

12.1.1.3 Compromises

- **Goal**: Breaking into the system and gaining privileged access to hosts.
- **Strategies**: Use known bugs and vulnerabilities.

Examples of compromises:

- R2L - Remote to local attacks, where an attacker who has the ability to send packets to a machine over a network (but does not have an account on that machine), gains access (either as a user or as the root) to the machine; usually based on password guessing or known vulnerabilities (e.g buffer overflows in Sendmail).
- U2R - User to root attack, where an attacker who has an account on a computer system is able to misuse/elevate her or his privileges by exploiting a vulnerability in computer mechanisms, a bug in the operating system or in a program that is installed on the system.

12.1.1.4 Viruses/Worms/Trojan horses/RootKits

- **Goal**: various + replicate on host machines and propagate through a network.
- **Strategies**:
  - Viruses: Programs that reproduce themselves by attaching them to other programs and infecting them. Typically need human interaction for replication and spreading to other computers.
  - Worms: Self-replicating programs that aggressively spread through a network. Can be categorized depending on the medium used for diffusion: traditional (direct network connection), e-mail (or other client applications), Windows file sharing protocols, hybrid ([16]).
  - Trojan horses: Malicious, security-breaking programs that are disguised as something benign. Typically the user download and activates the attack on purpose.
  - RootKits: Piece of software that once installed on a victim's machine opens up a port to allow a hacker to communicate with it and take full control of the system (back door). Some root kits give a hacker even more control of a machine than a victim may have themselves.

12.1.1.5 Man-in-the-Middle

- **Goal**: intercept communication to gather confidential data or inject false information.
- **Strategies**: The attack undergoes several steps:
  - scanning and eavesdropping
  - intrusion on a connection
  - message interception
  - selective data modification

One way to mitigate this type of attack is by encrypting communications with strong protocols. However, there are environments where these attacks are still very effective, such as IoT environments as they cannot implement encryption systems. That said, it is useless to monitor these types of attacks in areas of the network where encryption is implemented.

12.1.2 Involved Network Connections

- **Single connection**: To conduct the attack we can also be alone as we only need one point of contact with the network. Typical of MITM attacks.
- **Multiple connections**: To conduct the attack we must mask our presence in the network (as happens with Probing/scanning attacks) or increase the number of damages (for example in a DoS attack).

12.1.3 Attack Source

- **Single Source**: Typically, in this type of attack, we consider a single attack using multiple connections.
- **Multiple Source**: Usually, in this case, we want to make it more difficult to detect the intrusion within the system (we mask the alerts that the analyst sees), as we lighten the load of suspicious resources on each point used for the attack and we make the detection phase itself slower.

12.1.4 Environment

- **Host machine**: If a single host is attacked, we must be attentive to all events related to that host and in this case they are information on the processes that are running, consumption of resources, the pattern of processes performed.
- **Network**: If the attack occurs at the network level, we must detect as much information as possible at the level of interactions between the elements of the network itself. We must therefore see the number of connections, the type of traffic, the content of the packets, etc. We will not monitor the machine, but its interfaces.
- **P2P-like**: In this case, the hosts are all on the same level, so we focus on both the communications between machines and the processes they run.
- **Wireless Network**: Communications don't take place in P2P sessions, but we typically rely on broadcast communications and in this case we need to monitor nearby interactions.

12.1.5 Automation Level

- **Automated attacks**: Use automated tools that are capable of probing and scanning a large part of the Internet in a short time period.
- **Semi-automated attacks**: Deploy automated scripts for scanning and compromise of network machines and installation of attack code, and then use the handler (master) machines to specify the attack type and victim's address.
- **Manual attacks**: Involve manual scanning of machines and typically require a lot of knowledge and work. They are not very frequent, but they are usually more dangerous and harder to detect than others.

12.2 General framework

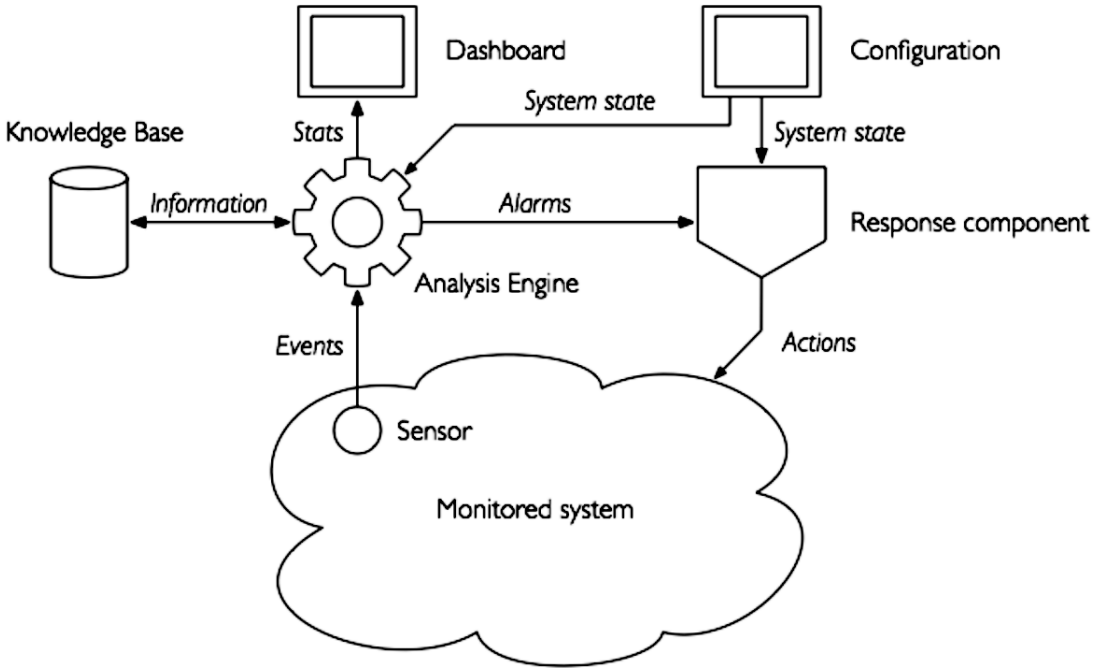


Figure 31: General framework of Intrusion Detection Systems.

As shown in Figure 31 we have the following elements that make up an IDS:

- **Monitored system**: The infrastructure we want to protect and monitor to detect suspicious activity.
- **Sensors**: They acquire information from the system and send it to the analysis engine. Work in event-based mode.
- **Analysis engine**: It collects all the events received by the sensors, analyzes them and, during this phase, acquires information from a possible knowledge base. Compare information from the knowledge base and compare it to system events.
- **Knowledge base**: This can be internal or external to the IDS and contains information related to typical attacks or system behaviour.
- **Dashboard**: High-level information is displayed here and made available to analysts, who must interpret behaviours in the system and, where necessary, issue an alarm.
- **Response component**: If our defense includes a malicious event response component (IPS), it collects the relevant warning, interprets it, and acts in some way on the system to promptly attack the threat.
- **Configuration**: The responses to be implemented in the system are defined in advance by the security operator.

12.2.1 Desired characteristics for an IDS

- **Detection Accuracy**: Reduce false positives so that no alarms are raised and resources wasted triggering unnecessary countermeasures. Reduce false negatives, which reduces the risk of not detecting attacks.
- **Timeliness**: Minimize the intrusion detection time.
- **Robustness**: IDSs are vulnerable themselves, so we must protect them too.

12.3 Taxonomy

There is a great variety of IDSs and therefore it makes sense to classify the most popular ones. We can group them all based on 5 macro-categories that correspond to peculiar characteristics (Figure 32).

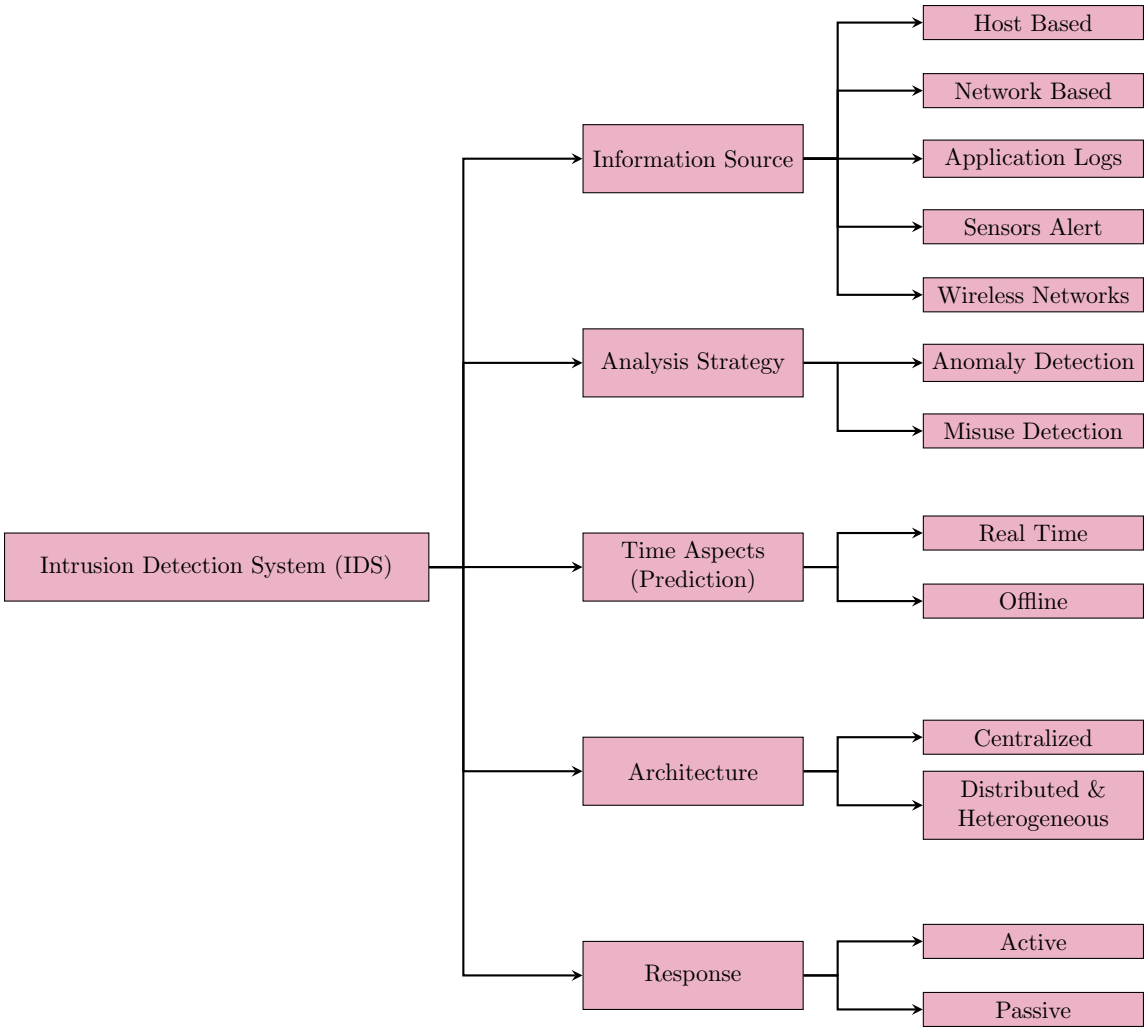


Figure 32: Intrusion Detection System (IDS) Taxonomy.

12.3.1 Information Source

12.3.1.1 Host based

The concept is to monitor events involving a specific host. Typically we collect information inside the machine and its interfaces, however, we will not explicitly consider the interactions, but more the flows it generates. In particular, we will analyze network traffic, log files, running processes, file access and configuration changes. Typical techniques used in these cases by IDSs are: code analysis, sandbox-based execution, traffic analysis, filesystem monitoring and log analysis. Typically this approach is used for the most delicate hosts of the company, such as a server or when we want to analyze data exchanged on encrypted channels.
CONS: The lack of context for detected actions make more difficult to detect attacks. Need complex configuration and extensive tuning and can have a negative impact on the performance of co-hosted applications.

12.3.1.2 Network based

The concept is to monitor traffic on a network segment, not only traffic at interfaces, but also traffic between hosts. Information can be analyzed at different levels of the ISO/OSI stack:

Application layer (HTTP, SMTP, POP/IMAP, DNS), Transport/Network layer (IP, TCP, UDP, ICMP) and Data-link layer (MAC, ARP). Deployment type can be:

- **Inline** where it allows the passage of traffic inside it, moreover, as for an IPS, we can block the traffic in a timely manner and therefore also the attacks.
- **Passive** where we duplicate the traffic (mirroring) and redirect it to the IDS, which will analyze it.

Typically the network-based IDS work in stealth mode, that is, they do not associate traffic with IPs, but only keep track of the interfaces and for this reason, they are very resistant to attacks, however, there is a high consumption of resources. Although it is very expensive in most cases it does not interfere with the flow of data in any way. IDSs of this type cannot monitor data passing through encrypted channels.

12.3.1.3 Application Logs

Monitor only specific applications such as database management systems, content management systems, accounting systems, etc. In this case, the IDS has access to information that host-based and network-based IDS cannot access. It can keep track of session information, however, this type of IDS is difficult to set up because we don't know in detail what is the granularity of the logs taken into consideration.

12.3.1.4 Sensors Alert

They are sources of information that come from software sensors that provide data collected from the environment. Typically in this case the IDS is hierarchical-based or graph-based topology.

12.3.1.5 Wireless Networks

The wireless medium offers new possibilities but also new risks. The physical layer in wireless networks is essentially a broadcast medium and therefore less secure than wired computer networks, there is no routing in the network and this implies that the communication is based on a sort of quorum and therefore this absence of routers, which convey on they traffic, makes it difficult to position the sensors; this problem is also because a wireless network does not have well-defined boundaries. We only have devices and applications so it becomes difficult to separate the malicious traffic from the legitimate one.

12.3.2 Analysis Strategy

12.3.2.1 Anomaly Detection

Strategy based on knowledge about the system, in this case, the focus is on identifying the anomaly with respect to normal behaviours. Everything that departs from the model is a potential attack.

- **Programmed systems**: the system is configured with fixed behavioural models. Default deny: the system expected behaviour is accurately model, so only modelled states are allowed. Descriptive statistics: the normal behaviour of the system is described by a statistical model built on a number of variables.
- **Self-learning systems**: build automatically a model representing the system normal behaviour. Non-time series: use stochastic modelling that does not consider time. Time-series: the model takes into account the time correlation between events.
- **Rule based methods**: characterize normal behaviour of users, networks and/or computer systems by a set of rules, when rules are broken, an attack is suspected.
- **Statistical methods**: monitor the user or system behaviour by measuring certain variables over time, then keep averages of these variables (moving event/time windows) and detect whether thresholds are exceeded based on the standard deviation of the variable.
- **Distance based methods**: characterize the normal behaviour through a vector and measure the geometric distance between this vector and the one that represents the observations, so provide a measure of the deviation.
- **Profiling methods**: a profile characterizing the normal execution of protocols and services is generated. Any deviation from the profile is considered as suspicious.

12.3.2.2 Misuse Detection

Strategy based on knowledge of previous attacks. We have the attack model and what we are going to do is define the incorrect behaviour or the signature of the attack. The IDS will ignore all normal behaviours and will only look for those that match the collected signatures. Everything works fine when we have the attack paths and we can immediately recognize the incorrect behaviour, but if we were to have even a small deviation from the signature, this analysis strategy would fail.

- **Signature-based IDSs**: Search a database containing "fingerprints" of known attacks, works like antivirus software, but cannot detect new types of attacks and has difficulty detecting old attack variants, which is why must keep the signature database up-to-date. An example is SNORT.
- **Rule-based**: Performs matching by evaluating "if...then" conditions. For each type of attack, we have a set of rules that should be checked and depending on which ones we have we can classify the set of events we are observing into one or more attack classes.
- **State transition analysis**: Requires the construction of a finite state machine where each state corresponds to different states of the network protocol stacks or to the integrity and validity of currently running processes, etc. When the automaton reaches a state that is flagged as a security threat, the intrusion is reported as a sign of malicious attacker activity.
- **Machine-learning based techniques**: Each instance in a data set is labelled as normal or intrusive and a learning algorithm is trained over the labelled data. It has a high degree of accuracy in detecting known attacks and their variations concerning signature-based intrusion detection systems.

12.3.3 Time Aspects

12.3.3.1 Real Time

The data is analyzed online, i.e. they take a data stream as input and analyze it at runtime. Useful for detecting threats early and reacting promptly. Cannot work on events that are produced out of sync.

12.3.3.2 Offline

The data is analyzed after being collected. In this case, performance is rarely a problem and also can work on more comprehensive datasets.

12.3.4 Architecture

12.3.4.1 Centralized

Data analysis is performed at a fixed number of locations, regardless of the number of hosts monitored. We are only interested in the aspects of analysis, not data collection, this simplifies configuration and management, but makes the method less fault-tolerant and less scalable with respect to the load.

12.3.4.2 Distributed & Heterogeneous

The analysis of the data is performed in several locations that are proportional to the number of hosts being monitored, this arises a complex configuration and management, but easier customization.

12.3.5 Response

Reaction types:

- Typically IDSs only report alarms to human administrators.
- Non-destructive reaction operations can be employed for specific attacks: patching and firewall rule injection.
- The most common reaction is an increase in the sensitivity of sensors to gather more detailed information.

13 Incident Management

Definitions:

- An **incident** is an event that could lead to loss of, or disruption to, an organization's operations, services or functions.
- **Incident management** is the process aimed at identifying, analysing, and correcting hazards to prevent a future re-occurrence. Incident management seeks to prevent such incidents from happening. When they do happen, incident management aims to contain and resolve them, and use the lessons learnt for the next time.
- A **Cyber Security Incident** is any malicious act (or suspicious event) that compromises (or was an attempt to compromise) the Electronic Security Perimeter or Physical Security Perimeter of a Critical Cyber Asset or disrupts (or was an attempt to disrupt) the operation of a Critical Cyber Asset.

For **NIST** (SP 800-61) an **incident** is the act of violating an explicit or implied security policy [17]. These include but are not limited to:

- attempts (either failed or successful) to gain unauthorized access to a system or its data;
- unwanted disruption or denial of service;
- the unauthorized use of a system for the processing or storage of data;
- changes to system hardware, firmware, or software characteristics without the owner's knowledge, instruction, or consent.

The incident management process is divided into 6 phases, as we can see from Figure 33.

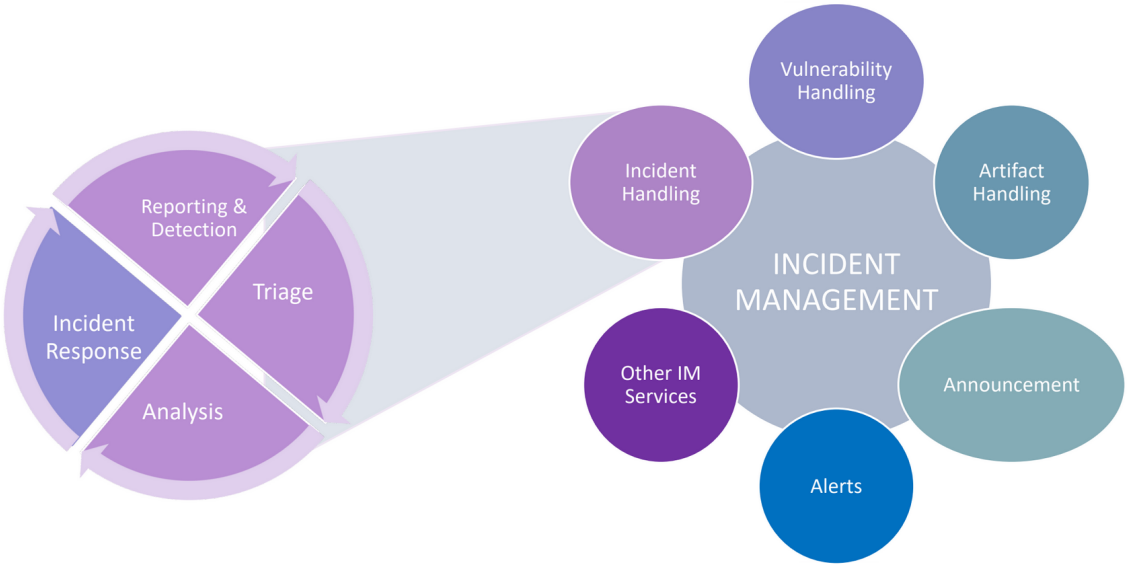


Figure 33: Incident Management Decomposition.

The central part of the process is however that of Incident handling, which can be divided into a cycle:

- **Detection & Reporting**: we detect the incident and report it to the security analyst.
- **Triage**: the analyst tries to identify its severity (prioritization), then tries to understand if the incident can be ignored or must be managed.
- **Analysis**: analyzes the evolution of the incident.
- **Incident response**: we plan an adequate response to the incident under analysis.

These 4 phases of the incident handling cycle can be seen as an integral part of a life cycle (Figure 34) that is composed by:

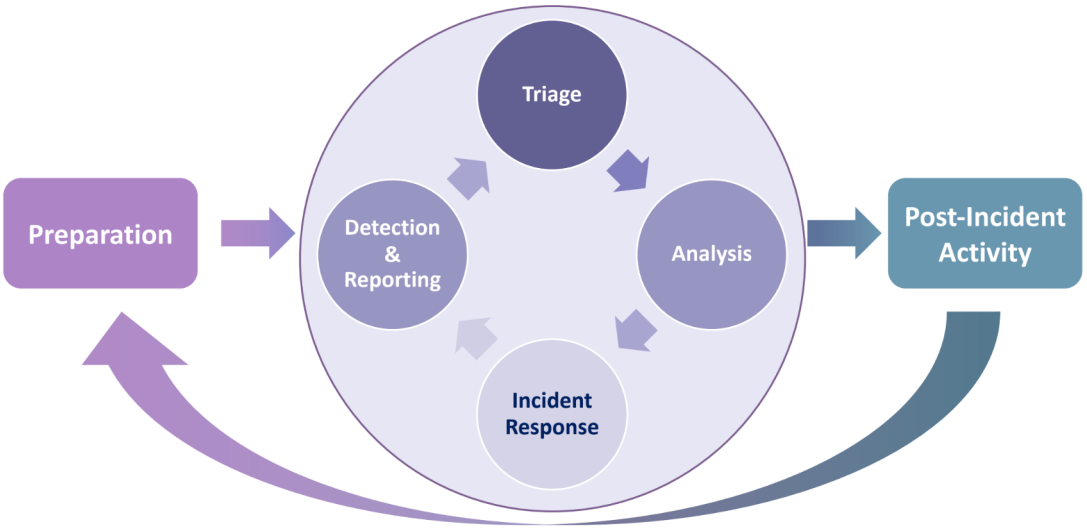


Figure 34: Incident Handling Life Cycle.

- **Preparation**: Typically includes all activities aimed at preventing the incident and preparing the structure to react (detect and react activities).
- **Incident handling loop**: The preparation activities give life to the incident handling loop, therefore: detection, triage, analysis and incident response. Note that after responding to the incident, we continue to observe the environment to review the situation and assess whether the response given was effective in resolving the incident or we have only mitigated it and we must continue to resolve it. And we continue with the loop until the incident is resolved.
- **Post-incident analysis**: Once the dangerous situation is over we start with the post-incident analysis. This analysis will provide us with feedback on the causes of the incident, the techniques used to bring down the incident. Based on what is reported in the analysis, we can decide whether to change things in the system (long-term countermeasures) or we can change the steps related to the preparation phase.

The incident handling life cycle is a particular instance of the direct-control loop that aims to improve the general process of incident handling and therefore Incident Management.

13.1 Preparation

Incident response methodologies typically emphasize preparation establishing an incident response capability to timely respond to incidents, but preventing incidents by ensuring that systems, networks, and applications are sufficiently secure is also paramount.
Preparation is composed of two main activities:

- **Preparing to handle Incidents**: In order to handle incidents efficiently and effectively there is the need to define:
  - **Communication and Facilities** that is, the mechanisms by which incidents are reported, and structures/facilities as contact Information, Incident reporting mechanism, Issue tracking system, etc.
  - **Analysis Hardware and Software tools**: Digital Forensics workstation, Laptops, Removable media, etc.
  - **Incident Analysis Resources**: Documentation, cryptographic hashes, list of critical assets.
  - **Incident Mitigation Software**: Clean OS and application for restoration.

  Many incident response teams create an emergency kit which is a portable case that contains materials that may be needed during an investigation. The jump kit should always be ready to use and includes: a laptop, loaded with appropriate software (e.g., packet sniffer, digital forensics), backup devices, blank media, and basic networking equipment and cables. Since

the purpose of having a jump kit is to facilitate quicker responses, the team should avoid borrowing items from the jump kit.

- **Preventing Incident**: Keeping the number of incidents that occur low is crucial because the higher the number of incidents within the system, the greater the likelihood that the response team will not be able to resolve all problems promptly. Therefore some fundamental activities must be put in place to prevent most incidents and among these we find:
  - Risk assessment
  - Host security
  - Network security
  - Malware prevention
  - User awareness and training

13.2 Detection and Analysis

Developing a step-by-step guide to managing incidents is not the smartest possible solution as an incident can occur in many different ways, so it would be better to manage incidents starting from the identification of their common attack vectors: external/removable media, attrition (e.g. brute force attacks), web etc. For this reason, it is crucial to be able to detect and classify incidents by looking at two important "signs":

- **Precursors** are those signs that warn us that an incident can happen in the future.
- **Indicators** are those signs that an incident may have occurred or may be occurring now.

13.2.1 Sources of Precursors and Indicators

Common Sources of precursors and indicators are shown in Table 28.

Alerts	IDPSs
	SIEMs (Security Information and Event Management)
	Antivirus and antispam software
	File Integrity Checking software
	Third party monitoring services
Logs	Operating system, service and application logs
	Network device logs
	Network Flows
Publicly Available Information	Information on new vulnerabilities and exploits
People	People from within the organization
	People from other organizations

Table 28: Common Sources of precursors and indicators.

13.3 Incident Analysis

Incident detection and analysis is affected by several factors such as accuracy (False Positive), a huge amount of events/alerts to analyze, indicators may follow from different root causes (not necessarily related to an incident) and many incidents are not associated with clear symptoms. The best thing to do is build a team of highly experienced and proficient staff members who can analyse the precursors and indicators effectively and efficiently and take appropriate actions. For this reason, there are 10 recommendations for Incident Analysis:

1. Profile networks and systems

2. Understand normal behaviours
3. Create a log retention policy
4. Perform event correlation
5. Keep all host clocks synchronized
6. Maintain and use a knowledge base of information
7. Use internet search engines for research
8. Run packet sniffers to collect additional data
9. Filter the data
10. Seek assistance from others

In case of suspect of an incident running, it is fundamental to start to record facts related to the possible incident:

- Timeline
- Supporting tools should be adopted
- Preserving integrity and confidentiality of collected data is important

13.4 Incident Prioritization

In the end, incident handling should be prioritized during triage, typically, based on the main factors such as:

- Functional Impact of the Incident
- Information Impact of the Incident
- Recoverability from the Incident

The classification and prioritization of incidents are essential for an organization, because resources are optimized if these are limited.

13.5 Incident Notification

Incidents should be notified to the appropriate individuals so that all who need to be involved in the response will play their roles. This should be supported through notification list and procedures that should be specified in security policies and through multiple communication strategies which should be defined as fault-tolerant.

13.6 Choosing a Containment Strategy

This is one of the most important procedures as it allows us to stop the incident, and therefore stop the waste of resources and block the growth of damage. In some cases, it serves to gain time to come up with a repair strategy. Since this is a decision-making process, developing a predetermined strategy is a great idea. In fact, organizations should define acceptable risks in dealing with incidents and develop strategies accordingly. Note that Containment Strategies are **incident dependent**. Criteria for determining the appropriate strategy include:

- Potential damage to and theft of resources
- Need for evidence preservation
- Service availability
- Time and resources needed to implement the strategy
- Effectiveness of the strategy
- Duration of the solution

13.7 Evidence Gathering and Handling

Gathering evidence during an incident has two main purpose:

- to resolve the incident
- for legal proceedings

In such cases, it is important to clearly document how all evidence (including compromised systems) has been preserved.

13.7.1 Identifying the Attacking Hosts

Identifying an attacking host can be a time-consuming and futile process that can prevent a team from achieving its primary goal i.e. minimizing the business impact. The most commonly performed activities for attacking host identification are:

- Validating the attacking host's IP address.
- Researching the attacking host through search engines.
- Using incident databases.
- Monitoring possible attacker communication channels.

13.8 Eradication and Recovery

Eradication may be necessary to eliminate components of the incident, disable breached user accounts and identify and mitigate all vulnerabilities that were exploited. In recovery, administrators restore systems to normal operation, ensure that systems are functioning normally, and remediate vulnerabilities to prevent similar incidents. The procedures involved in this last phase may include: restoring systems from clean backups, rebuilding systems from scratch, replacing compromised files with clean versions, installing patches, changing passwords and tightening network perimeter security.

13.9 Lessons Learned

This is one of the often omitted parts of the whole procedure, but one of the most important, namely the one that concerns **learning** and **improvement**. Questions to be answered include:

- Exactly what happened, and at what times?
- How well did staff and management perform in dealing with the incident? Were the documented procedures followed? Were they adequate?
- What information was needed sooner?
- Were any steps or actions taken that might have inhibited the recovery?
- What would the staff and management do differently the next time a similar incident occurs?
- How could information sharing with other organizations have been improved?
- What corrective actions can prevent similar incidents in the future?
- What precursors or indicators should be watched for in the future to detect similar incidents?
- What additional tools or resources are needed to detect, analyse, and mitigate future incidents?

14 How to support the Incident Management: SOC and CERT

14.1 Security Operation Centre (SOC)

A Security Operation Centre (SOC) can be defined as a **centralized** security organization that assists companies with **identifying**, **managing** and **remediating** distributed security attacks. Depending on the capabilities required from a SOC by the enterprise or client, a SOC can also be responsible for the management of technical controls. The end-goal of a SOC is to improve the security posture of an organization by **detecting** and **responding** to threats and attacks **before they have an impact** on the business [18]. The services offered by the SOC are:

- Log Management
- Security Monitoring and Alerting
- Security Incident Management
- Security Operation Management
- Vulnerability Assessment
- Service Security Assessment
- Security Analytics starting from data collected from SIEM
- Threat Intelligence (in partial overlapping with CERTs)

14.1.1 Building Blocks of a SOC

Figure 35: Triad of Security Operations: People, Process and Technology from [19].

SOC is at the heart of 3 elements (Figure 35): People, Process and Technology. Remind that when we want to create an efficient SOC that delivers effective services we cannot ignore these 3 elements, thus selecting the right people (formal training, on-the-job experience, vendor-specific training and internal training), the right processes (preparation, identification, containment, eradication, recovery and lessons learned (Section 13.9)) and the right technologies (endpoint, NetFlow, network monitoring, Threat Intelligence, forensics and incident detection/management) is crucial.

14.1.2 Organization of the SOC

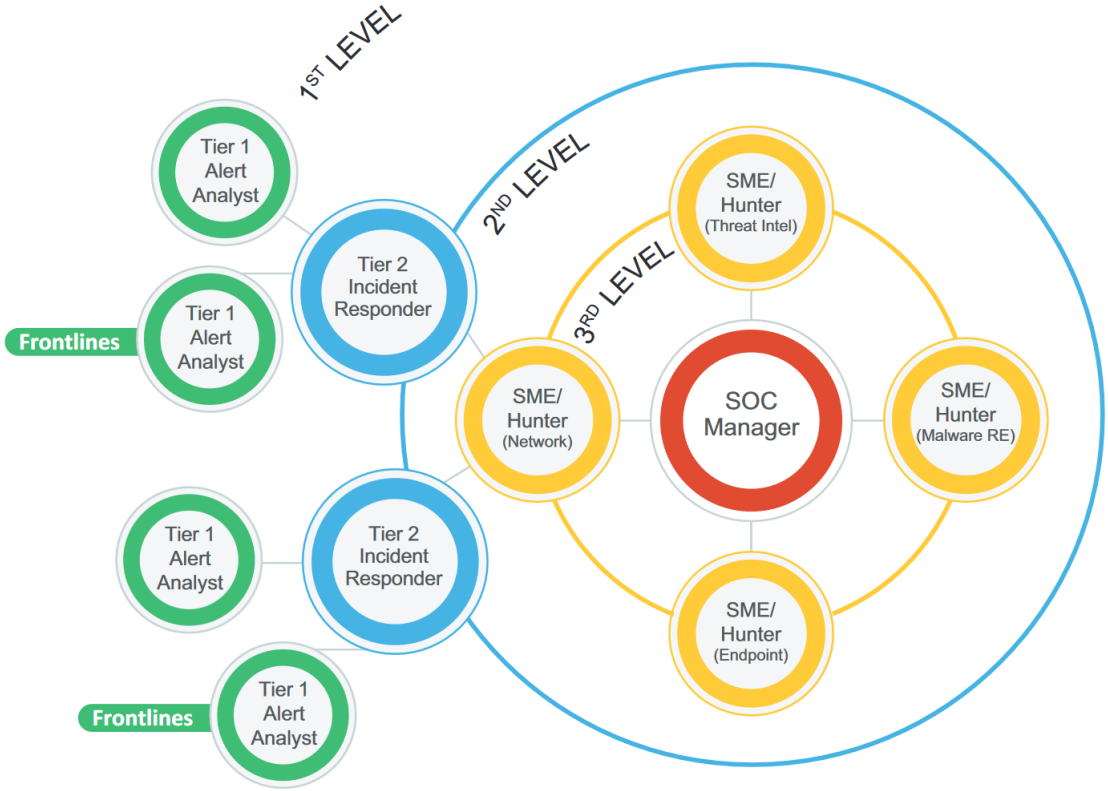


Figure 36: Organization Chart of Security Operations Center from [19].

The SOC has 3 levels, above which there is the SOC Manager, who is responsible for all SOC activity. There is a hierarchical structure as can be seen from Figure 36: the strategic layer is that of the SOC manager, the tactical layer is between the 2^{nd} level and the 3^{rd} level, finally, the 1^{st} level is equivalent to the operational layer. Let's see them in more detail:

- **Tier 1 - Alert Analyst**: Continuously monitors the alert queue; triages security alerts; monitors health of security sensors and endpoints; collects data and context necessary to initiate Tier 2 work.
- **Tier 2 - Incident Responder**: Performs deep-dive incident analysis by correlating data from various sources; determines if a critical system or data set has been impacted; advises on remediation; provides support for new analytic methods for detecting threats.
- **Tier 3 - Expert Security (Threat Hunter)**: Possesses in-depth knowledge on network, endpoint, threat intelligence, forensics and malware reverse engineering, as well as the functioning of specific applications or underlying IT infrastructure; acts as an incident "hunter", not waiting for escalated incidents; closely involved in developing, tuning and implementing threat detection analytics.
- **SOC Manager**: Manages resources to include personnel, budget, shift scheduling and technology strategy to meet Service Level Agreements; communicates with management; serves as organizational point person for business-critical incidents; provides overall direction for the SOC and input to the overall security strategy.

Hence, the SOC is concerned with collecting data from heterogeneous sources and implementing a selection and correlation (Figure 37).

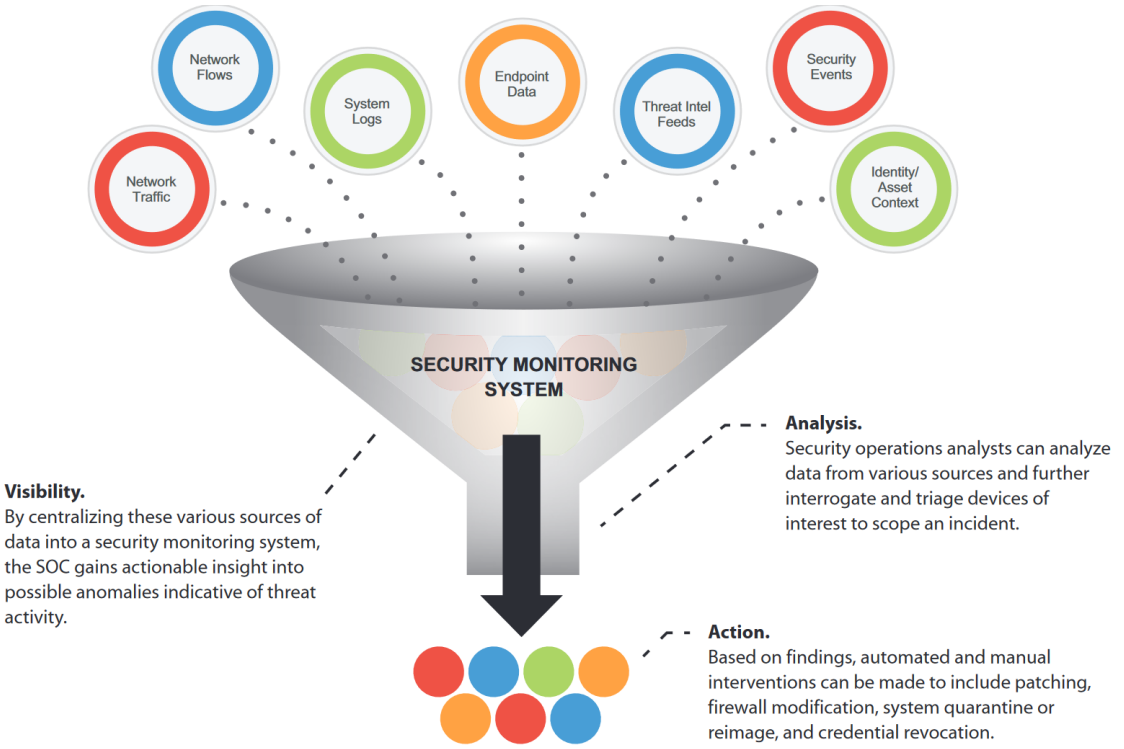


Figure 37: Data Aggregation for Improved Incident Handling from [19].

14.1.3 When you should adopt a SOC

It is good to have a SOC in presence of critical or "sensitive" data or processes (in terms of business and/or regulatory compliance), there is a growing trend of the company and an internal information security function (not structured and/or strongly unbalanced on the work of an outsourcer) can no longer "keep up" or there is the need to equip themselves with "pushed" monitoring and technical response capacities to information security events.

14.2 Computer Emergency Response Team (CERT)

A **Computer Emergency Response Team (CERT)** is a group of experts who respond to cybersecurity incidents. Their main functions are:

- Providing preventive services (such as alerts on cyber security attacks)
- Providing security bulletins (advisory)
- Training
- Providing management of security services (function in overlap with a SOC)

Nowadays, however, we also have another definition for the CERT: **Computer System Incident Response Team (CSIRT)**, that is, that team specialized in responding to incidents related to computer security. Currently, CERT and CSIRT are used as synonyms.
A framework for building up a successful CERT/CSIRT is based on these foundations:

- **Mission** why the CERT was created and what is the set of activities it will support.
- **Constituency** is the organization or set of organizations that asked us to manage the incident.
- **Responsibility** what the CERT is supposed to do and what its action limit is.
- **Mandate** what is the power of the CERT with respect to the activities of the organization.
- **Organizational framework**
- **Available Service**

14.2.1 Responsibility

When defining responsibility, the following practical questions should be taken into account:

- What types of incidents must be handled by the CERT, and with what priorities?
- Must the CERT keep track of incident resolution and, at the end, close it? Or it is sufficient just to notify constituents to fulfil that task?
- Is the CERT obliged to actively solve an incident – which goes one step beyond guarding? Or just notify and give advice?
- Must the CERT escalate incidents when they do not get solved quickly enough and, if so, when and what must be escalated?
- Must the CERT inform specific entities about specific incidents? For example, when an employee may have done something "wrong", must the CERT inform its management, or the management of the employee, or the human resources department?
- Think through the CERTs responsibility by examining specific incidents. Was the responsibility clear enough? Where can it be improved or extended?

14.2.2 Mandate

When defining the mandate, the following practical questions should be considered:

- Does the CERT only give advice to its constituents, or can it also expect them to react in some way – such as giving acknowledgements, or even update reports – or can the CERT oblige them to solve the issue in a given time and keep the CERT informed?
- Can the CERT give deadlines to its constituents to solve incidents? If they do not meet that deadline, what sanctions can the CERT impose? Can the CERT isolate them from the internet or corporate network, or impose protocol specific filters? Can the CERT escalate and to whom, just to its own management, or also to the constituent's management?
- Can the CERT just provide co-ordination and advice regarding an incident, or can it also actively gather data in constituents' computers, possibly do forensics, etc?
- Think through the CERT's mandate by examining specific incidents. Is the mandate well defined? Where can it be improved or clarified?

14.2.3 Organisational Framework

Some aspects of governance are essential to good incident management and need to be thoroughly considered and clearly defined as escalation, relationship with CISO and CIO and relationship with Crisis Management inside organisation.

14.2.4 Services

Reactive Services	**Proactive Services**	**Security Quality Management Services**
• Alerts and Warnings • Incident Handling • Vulnerability Handling • Artifact Handling	• Announcements • Technology Watch • Security Audits or Assessments • Configuration and Maintenance of Security Tools, Applications and Infrastructure • Development of Security Tools • Intrusion Detection Services • Security-related Information Dissemination	• Risk Analysis • Business Continuity and Disaster Recovery Planning • Security Consulting • Awareness Building • Education Training • Product Evaluation or Certification

14.2.5 Roles

A CERT must include 4 mandatory roles:

- **Duty officer**: The person who must take charge of all incoming requests and must also carry out other periodic activities typical of his role.
- **Triage office**: The person who has to deal with all reported incidents decides if an incident should be managed by the team, when it should be managed and who will be the incident handler who will have to manage it. Finally, he must ensure a continuous updating of his notions regarding attack vector, trends and methods of attack used by the conspirators. Often the triage officer is also the duty officer.
- **Incident handler**: The person who directly faces events by analyzing data, creating workarounds that clearly communicate to clients the progress they achieve.
- **Incident Manager**: The person responsible for coordinating the incident handling activities.

The following roles are optional:

- Public relations officer
- Legal officer
- Team manager
- Hotline operator

However, in many cases part or all of the tasks that would fit the roles have to be undertaken in some way.

14.2.6 Incident Management Workflows

Figure 38 gives a general overview of each phase of the incident handling process. Before we decide on our detailed workflow, it is worth examining other examples of models and use them to develop our ideas, expectations and vision for our future model. Keep in mind that this workflow is close to our day-to-day operations and the workflow should enable our team to do the work more efficiently and not hinder the work. Below some further examples.

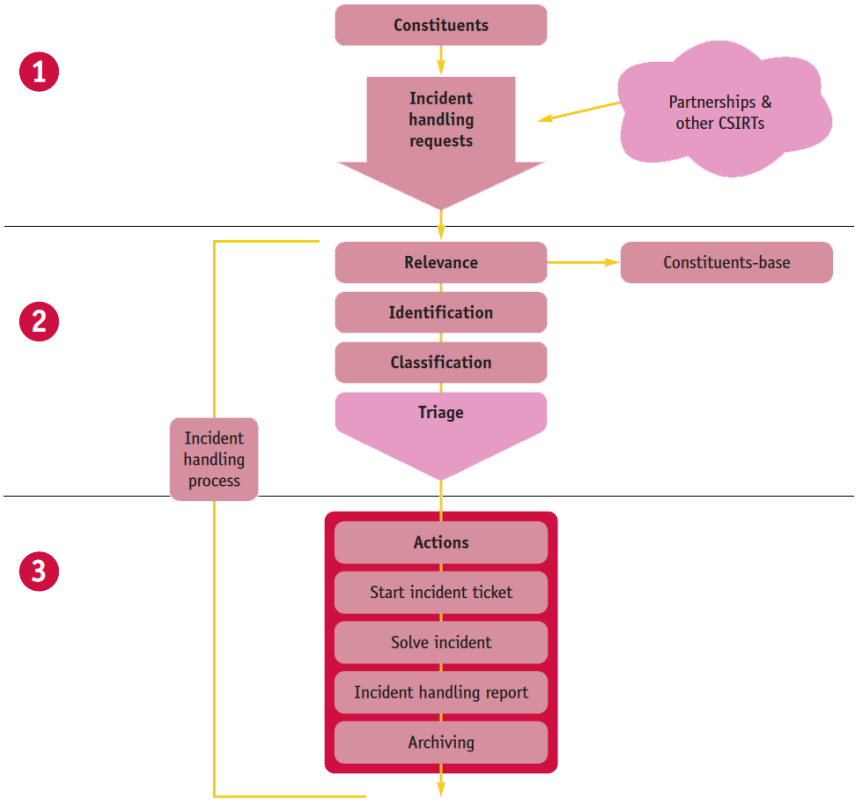


Figure 38: Incident handling process flow from [20].

Figure 39 represents the workflow used by CERT-Hungary - again derived from CERT/CC concepts. As we can see, everything starts with an incoming incident report followed by the triage process, which includes incident report confirmation. After this, the core part of the process (incident handling) starts. It consists of four phases in one cycle:

- Analysis
- Contact Information
- Technical Support
- Information and Response Coordination

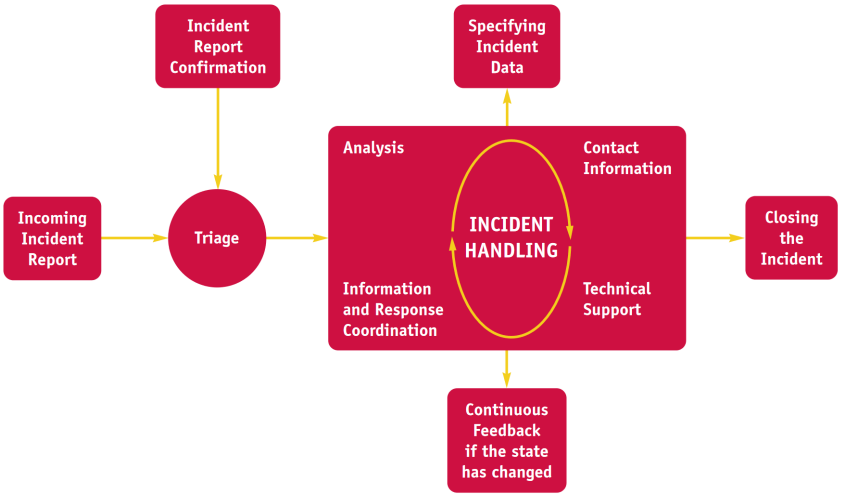


Figure 39: Incident handling process flow (CERT Hungary example) from [20].

Very important parts of this cycle are the continuous activities of specifying incident data and feedback between incident handling parties if the state has changed. All activities mentioned should lead to closing the incident. Therefore, as we can see, it is possible to extend the workflow and include other or more steps. Every step can be detailed with specific checklists. Developing them is a very educational process. The involvement of all our team members in this development can produce a common understanding of how our team works. It is very useful for self-learning on how to deal effectively with incidents. A good approach to the problem of the "exercise" is to prepare a general workflow that includes all the most important phases of the process as shown in Figure 40.

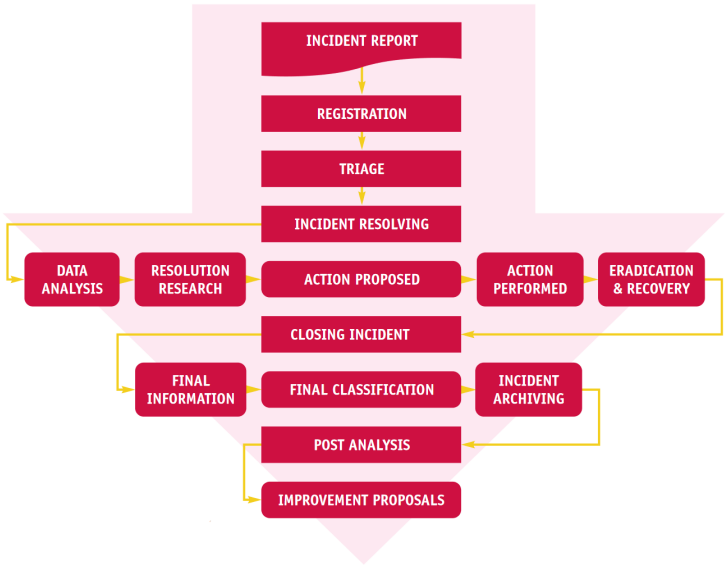


Figure 40: Incident handling workflow from [20].

According to our preferences, we can develop either a list of guidance and advice notes for an incident handler (fixed to the particular phases) or a more advanced workflow with graphical representation of decision trees (Figure 41).

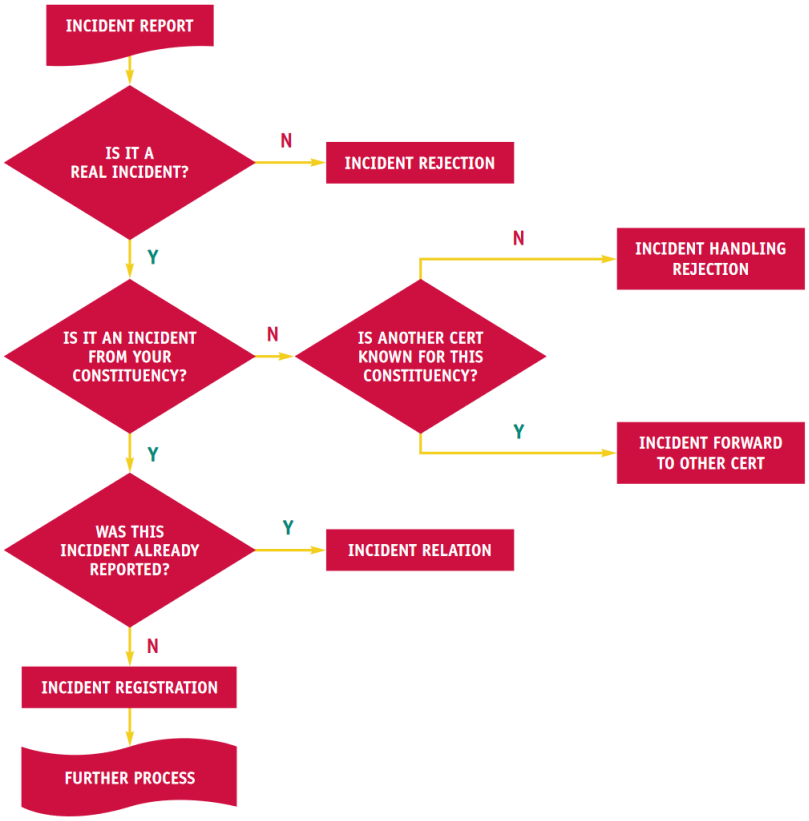


Figure 41: Part of a detailed incident handling workflow – graphical approach from [20].

14.2.7 Policies

A few policies are basic and independent of the CERTs' services or constituents, these are:

- Information Classification Policy
- Information Disclosure Policy
- Media Policy
- Privacy Policy
- Security Policy

14.3 Relationships between SOC and CERT

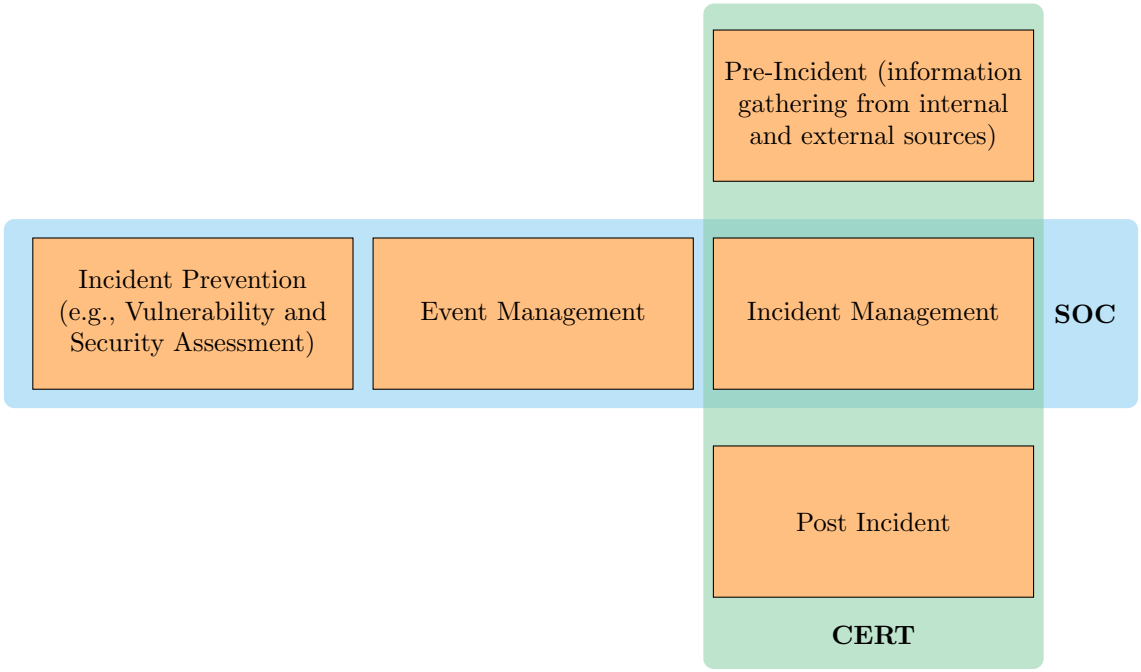


Figure 42: Graphical relationships between SOC and CERT.

SOC and CERT have a complementary perspective in the support they give to organizations regarding incident analysis and rejection. Although these have a common point which is the Incident Management process as can be seen from Figure 42, all SOC activities provide the organization with a solution completely dedicated (Incident Prevention), provide for Event Management, monitoring and analysis collection services and finally they support the organization in Incident Management. If we see the services provided by the CERT, we have to do with Pre-Incident information, where the CERT itself collects information from internal and external sources, but in this case, it is not the result of a detailed analysis tailored to the client, but a sort correlation of data between organizations to identify common problems, vulnerabilities and trends, this correlation will then be used for specific CERTs. Then we find Incident Management, in this case, how much the CERT is actively involved depends on what is decided with the clients, anyway it has to do it in direct communication with the SOC. Finally, we find the Post Incident in which the CERT collects information on the incident which, once again, will not be used for that specific client, but to increase the overall data and made available to all clients, to update trends, patterns attack etc.

15 Measuring Security and Security Metrics

We have two definitions from NIST for what security is:

- A **security attribute** is an abstraction representing the basic properties (or characteristics) of an entity with respect to safeguarding information [21].
- **Cybersecurity** consists in the prevention of damage to, protection of, and restoration of computers, electronic communications systems, electronic communications services, wire communication and electronic communication (including information contained therein) to ensure its availability, integrity, authentication, confidentiality, and non-repudiation [22].

15.1 Security Attributes

The security attributes are:

- **Confidentiality**: Absence of unauthorized disclosure of information.
- **Integrity**: Absence of unauthorized system alterations.
- **Availability**: Availability for authorized actions only.
- **Accountability**: Availability and integrity of the identity of the person who performed an operation.
- **Authenticity**: Integrity of a message content and origin, and possibly of some other information, such as the time of emission.
- **Non-repudiability**: Availability and integrity of the identity of the Non sender of a message or of the receiver.

15.2 Security Metrics

Depending on our analysis activity, there are many security metrics, some examples are to measure the direct control cycle where the risk metric is usually used or to measure system exposure, vulnerability-based metrics are usually used. We will focus on System Security Metrics, and most of the metrics presented here are based on the Attacker-Defender security approach:

- An **Attacker** is an entity representing a computer or an IP address from which cyber attacks are launched against other normal entities.
- An **Incident** represents a successful attack (e.g. malware infection or data breach).

15.3 Attack-Defence

15.3.1 Interactions in an Enterprise System

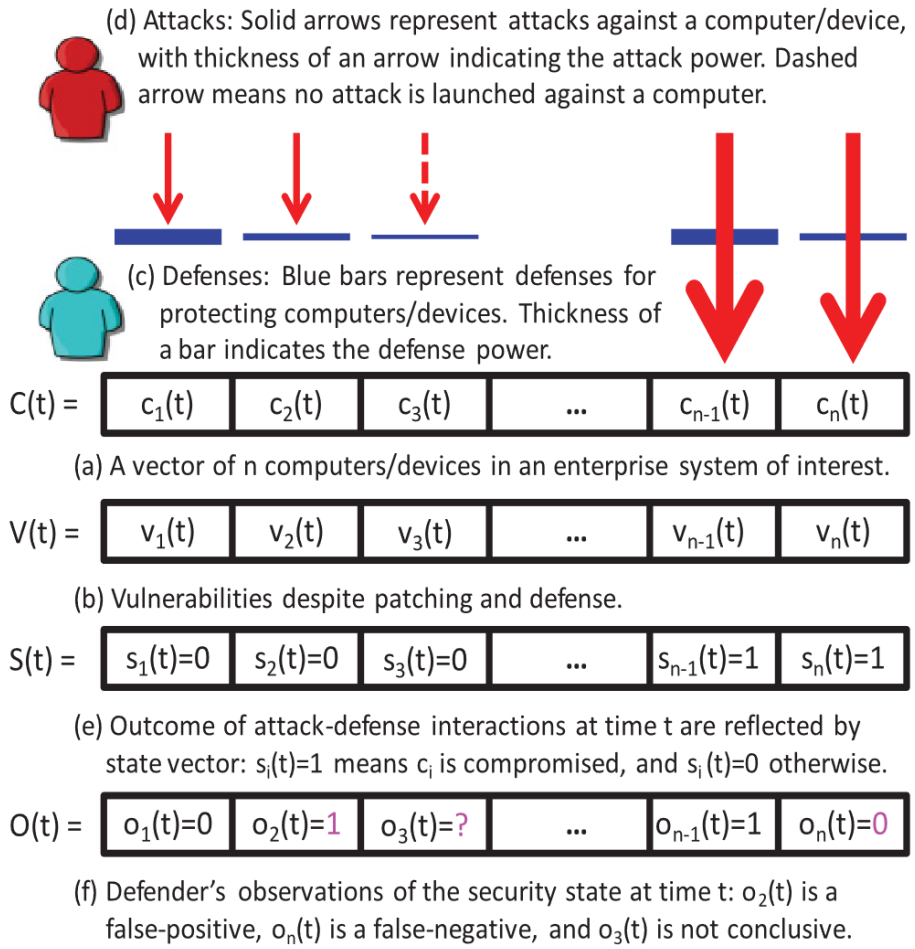


Figure 43: Attack-defence interactions in an enterprise system at time t from [23].

Figure 43 illustrates a snapshot of an enterprise system under attack-defence interactions. At time t, the enterprise system consists of n entities (i.e. computers), denoted by the vector $C(t) = c_1(t), \ldots, c_n(t)$, where n can vary over time t. Each entity, $c_i(t)$, has a vector $v_i(t)$ of vulnerabilities, such as zero-day and/or some unpatched software vulnerabilities. Red arrows refer to attacks and blue bars represent defences, where defence mechanisms are placed at both individual entities (e.g. anti-malware tools) and an enterprise system (e.g. firewalls). The thickness of red arrows and blue bars illustrates the intuitive notion of strength of attack and defence. Some attacks penetrate through the defences while others fail. The outcome of the attack-defence interaction at time t is reflected by a global security state vector $S(t) = s_1(t), \ldots, s_n(t)$, where $s_i(t) = 0$ means entity $c_i(t)$ is secure at time t and $s_i(t) = 1$ means entity $c_i(t)$ is compromised at time t. However, the defender's observation of the security state vector $S(t)$, denoted by $O(t) = o_1(t), \ldots, o_n(t)$, is imperfect due to detection errors or inherent noises.

15.3.2 Interactions in a Computer (or Device)

An example scenario is illustrated in Figure 44 where an attacker can perform a vector of 11 attacks, A_i for $i = 1, \ldots, 11$, some of which may successfully penetrate through the defence of $c_i(t)$. We have the arrows from left to right which are reduced because there is a first barrier where a filter is applied, after which part of the attacks are detected by some detection system (e.g. IDPS), finally on the right we have the attacks that are successful on our device in relation to the type of vulnerability.

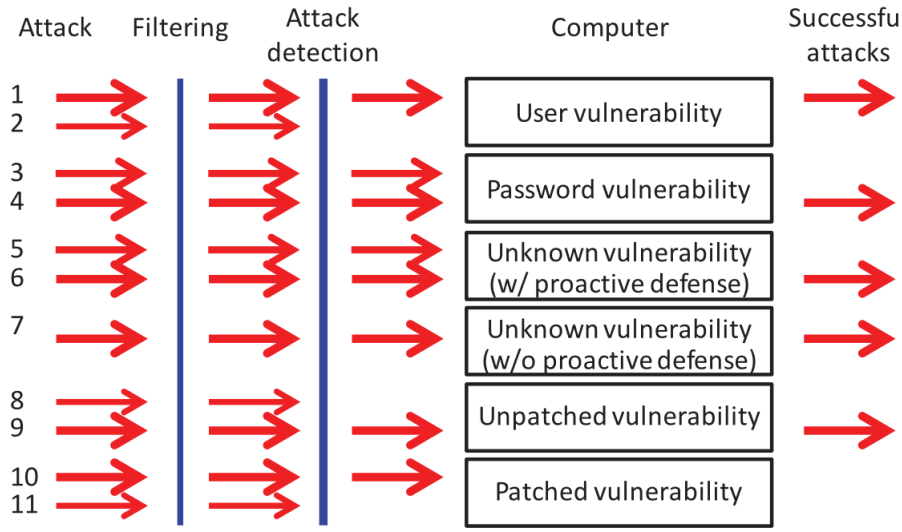


Figure 44: Attack-defence interactions in a computer (or device), $c_i(t)$, in the enterprise system at time t from [23].

15.3.3 Situation Understanding

The attack-defence interaction perspective leads to an intuitive formulation of security modeling. The outcome of attack-defence interaction is the evolution of the situation (or `situation(t)`) which is described by three classes of metrics:

$$situation(t) = f(V(t), D(t), A(t))$$

where:

- `V(t)` is a function of vulnerabilities at time t.
- `D(t)` is a function of defences at time t.
- `A(t)` is a function of attacks at time t.

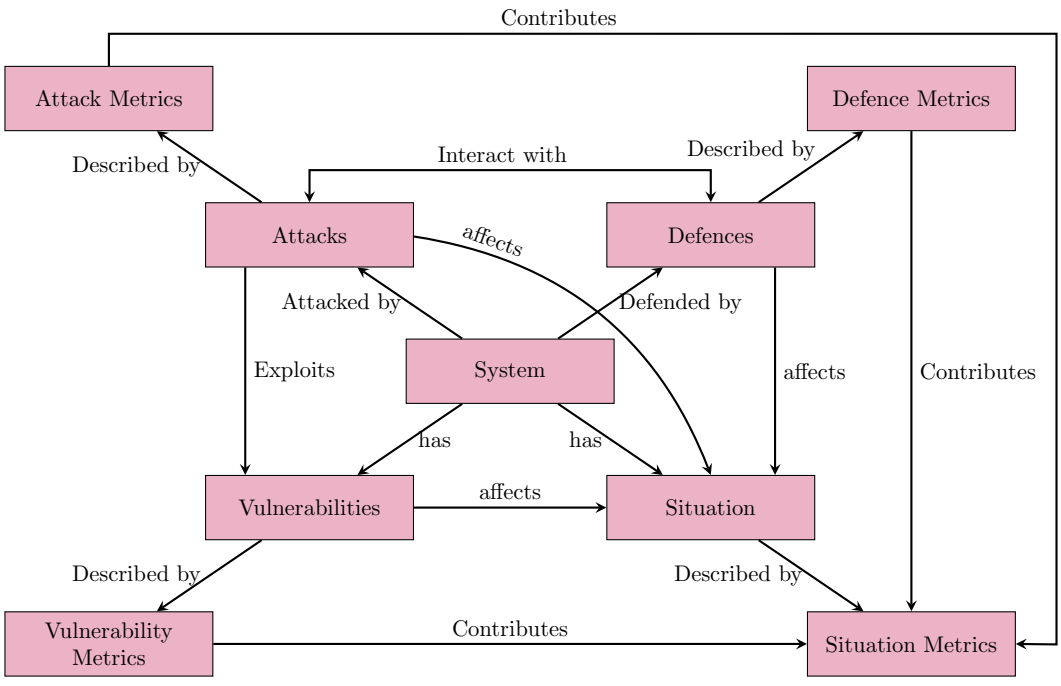


Figure 45: A high-level ontology of systems security metrics consisting of four metrics.

15.4 Vulnerability Metrics

Figure 46 shows the vulnerability metrics, let's take a closer look at each of these vulnerabilities below.

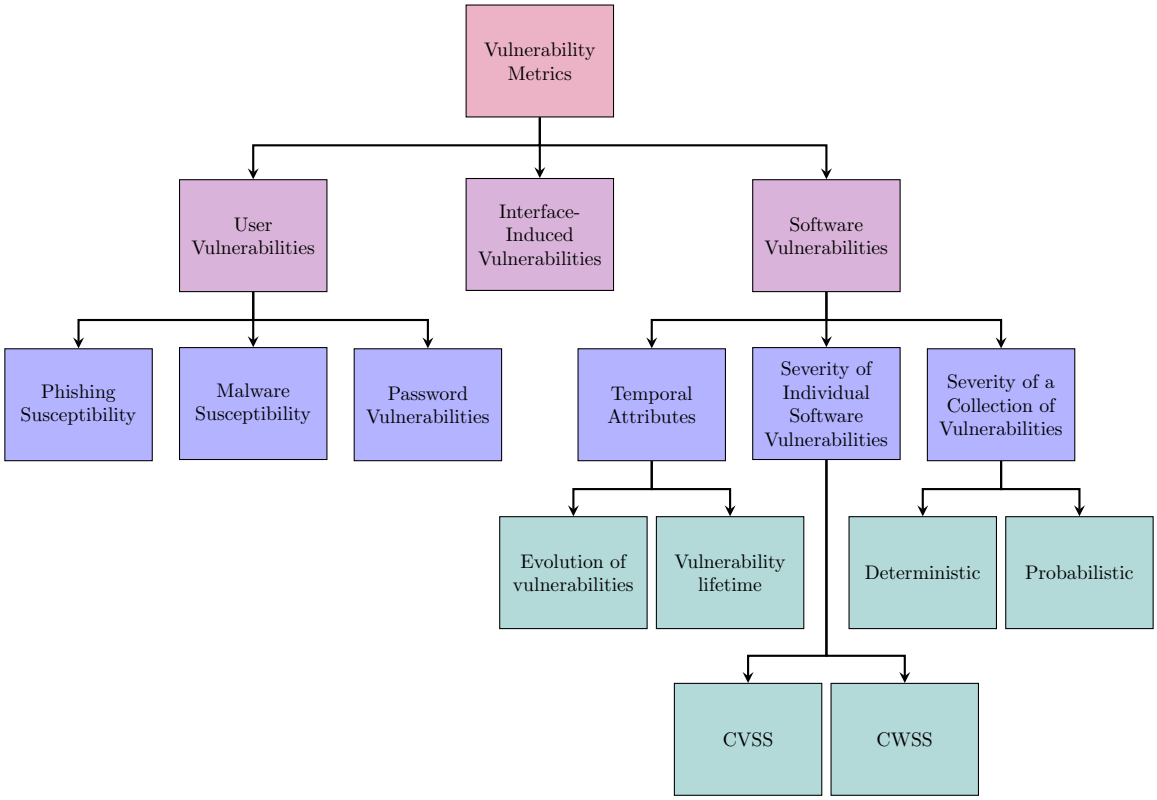


Figure 46: Vulnerability metrics.

15.5 Measuring User Vulnerabilities

Here we discuss two types of user vulnerabilities in terms of users' cognitive bias (or error) and users' cognitive limitation. A user's susceptibility to phishing attacks or insider threat is a typical example of vulnerabilities exposed by the user's cognitive bias, while weak password often happens due to limited human memory. Weak password can be easily broken by attackers, breaking an authentication mechanism.

15.5.1 Phishing Susceptibility

Typical metrics are false positives (FP) or false negatives (FN), where FP indicates the percentage of flagging genuine email as phishing email while FN captures the percentage of detecting a phishing email as a genuine email. Often human cognitive bias or personality traits can affect phishing susceptibility. The phishing susceptibility can be measured in a ratio scale. This type of test will be done without "warning" the user in order not to distort the statistics, or rather, they will only be warned that in a next period without specifying exactly when, their actions will be collected to create a statistic of awareness in the organization.

15.5.2 Malware Susceptibility

Malware susceptibility is closely related to a user's online behaviour. Users who often install many applications are more likely exposed to malware. In addition, if users visit many websites, then there is a higher vulnerability for malware infection. This malware susceptibility is also estimated in a ratio scale.

15.5.3 Password Vulnerabilities

Entropy is the most intuitive metric to measure the strength of a password, in fact, it is often estimated using heuristic rules and it offers a rough approximation of password weakness or strength and it cannot tell which passwords are easier to crack than others. Password guessability aims to measure password vulnerability via the time or number of guesses that a password-cracking algorithm takes to recover a password or a set of passwords, which is attractive because it is easy to use and can be used to compare password-cracking algorithms.

15.6 Measuring Interface-Induced Vulnerabilities

Attack surface metrics aim to measure the ways by which an attacker can compromise a targeted software. The attack surface of the software measures the subset of resources that can be abused to compromise the software, and it is defined as a tuple as follows:

$$\langle \sum_{m \in \mathcal{M}} R_{d,e}(m), \sum_{c \in \mathcal{C}} R_{d,e}(c), \sum_{i \in \mathcal{I}} R_{d,e}(i) \rangle$$

where $R_{d,e}$ represents the ratio of the potential damage and effort related to:

- The set of attack methods $\mathcal{M}$.
- The set of channels offered by the software $\mathcal{C}$.
- The set of data item of the software $\mathcal{I}$.

15.7 Measuring Software Vulnerabilities

We divide software vulnerability metrics into three categories: temporal attributes of vulnerabilities, individual vulnerabilities, and collective vulnerabilities.

15.7.1 Temporal Attributes

These metrics can be further divided into two sub-categories of metrics for measuring: the evolution of vulnerabilities and vulnerability lifetime.

15.7.1.1 Evolution of Vulnerabilities

Metrics for measuring the evolution of vulnerabilities include:

- **Historical vulnerability** measures the degree that a system is vulnerable (i.e.,frequency of vulnerabilities) in the past.
- **Historically exploited vulnerability** measures the number of vulnerabilities exploited in the past.
- **Future exploited vulnerability** measures the number of vulnerabilities that will be exploited during a future period of time.
- **Tendency-to-be-exploited** measures the tendency that a vulnerability may be exploited, where the "tendency" may be derived from information sources such as posts on Twitter before vulnerability disclosures.

15.7.1.2 Vulnerability Lifetime

Metrics for measuring vulnerability lifetime measures how long it takes to patch a vulnerability since its disclosure, measured by an absolute scale. The three vulnerabilities can be considered to measure different vulnerability lifetimes as:

- **Client-end vulnerabilities** are often exploited to launch targeted attacks (e.g. spear-fishing). It often takes a long time to patch all infected devices or possibly infeasible to patch all.
- **Server-end vulnerabilities** are usually more rapidly patched than client-end vulnerabilities.
- **Cloud-end vulnerabilities** have been reported by many clouds (e.g. Amazon), but the patching process is quite slow because there is a distributed system to patch and this leads to several problems (e.g. cost, time, etc).

15.7.2 Severity of Individual Software Vulnerabilities

The CVSS aims to measure software vulnerabilities with an emphasis on ranking them for prioritizing patching operations. The Common Weakness Scoring System (CWSS) aims to prioritize software weaknesses for a different purpose.

15.7.2.1 Common Vulnerability Scoring System (CVSS)

The Common Vulnerability Scoring System (CVSS) captures the principal technical characteristics of software, hardware and firmware vulnerabilities. Its outputs include numerical scores indicating the severity of a vulnerability relative to other vulnerabilities [24]. CVSS is composed of three metric groups:

- **Base metrics**: reflect the severity of a vulnerability according to its intrinsic characteristics which are constant over time and assumes the reasonable worst case impact across different deployed environments. These metrics are: Attack Vector, Attack Complexity, Privileges Required, User Interaction, Confidentiality Impact, Integrity Impact, Availability Impact and Scope.
- **Temporal metrics**: adjust the Base severity of a vulnerability based on factors that change over time, such as the availability of exploit code, remediation level and report confidence.
- **Environmental metrics**: adjust the Base and Temporal severities to a specific computing environment. They consider factors such as the presence of mitigations in that environment.

When the Base metrics are assigned values by an analyst, the Base equation computes a score ranging from 0.0 to 10.0.

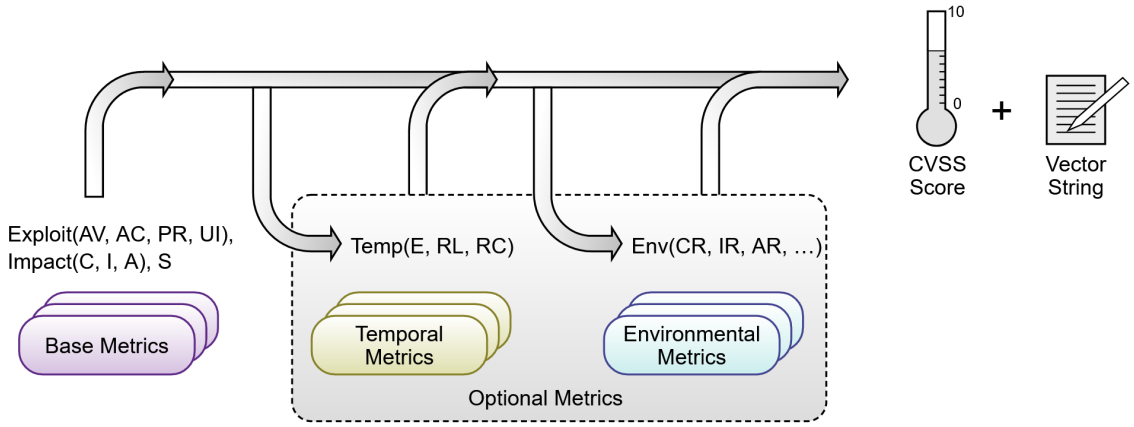


Figure 47: CVSS Metrics and Equations from [24].

15.7.2.2 Common Weakness Scoring System (CWSS)

The Common Weakness Scoring System (CWSS) provides a mechanism for prioritizing software weaknesses in a consistent, flexible, open manner. It is a collaborative, community-based effort that is addressing the needs of its stakeholders across government, academia, and industry [25]. CWSS is distinct from (but not a competitor to) the Common Vulnerability Scoring System (CVSS). Note that CVSS and CWSS have different roles, and they can be leveraged together. CWSS offers:

- **Quantitative Measurements** of the unfixed weaknesses that are present within a software application.
- **Common Framework** for prioritizing security errors ("weaknesses") that are discovered in software applications.
- **Customized Prioritization**

CWSS is organized into three metric groups:

- **Base Finding** metric group: captures the inherent risk of the weakness, confidence in the accuracy of the finding, and strength of controls.

- **Attack Surface** metric group: the barriers that an attacker must overcome in order to exploit the weakness.
- **Environmental** metric group: characteristics of the weakness that are specific to a particular environment or operational context.

To compute the score there are these steps:

1. Each factor in the Base Finding metric group is assigned a value, where those are converted to associated weights, and a Base Finding sub-score is calculated. The Base Finding sub-score can range between 0 and 100.
2. The same method is applied to the Attack Surface and Environmental metric group, and their sub-scores can range between 0 and 1.
3. The three sub-scores are multiplied together to get a CWSS score between 0 and 100.

15.7.3 Severity of a Collection of Vulnerabilities

Many attacks in the real world are performed in multiple steps and exploit multiple vulnerabilities. The most common methods for analysing collection of vulnerabilities are attack graphs, attack trees and privilege trees. We will consider two types of metrics measuring the severity of a collection of vulnerabilities (particularly useful in attack graphs) deterministic and probabilistic.

15.7.3.1 Deterministic Severity Metrics

These metrics are mainly defined over attack graphs. Two sub-metrics are:

- **Topology** metrics measure how the topological properties of attack graphs affect network attacks. We consider mainly two types of topology metrics:
  - **Depth** metric refers to a ratio of the diameter of a domain-level attack graph over the diameter in the most secure case, implying that the larger the diameter, the more secure the network.
  - **Existence, number, and lengths of attack paths** metrics use the attributes of attack paths from an initial state to the goal state.
- **Effort** metrics capture the degree of effort by a defender to mitigate vulnerability exploitation by attackers or by an attacker to exploit a given vulnerability. The common metrics under this metric category are:
  - **Necessary defence** estimates a minimal set of defence countermeasures necessary for thwarting a certain attack.
  - **Effort-to-security-failure** measures an attacker's effort to reach its goal state.
  - **Weakest adversary** estimates minimum adversary capabilities required to achieve an attack goal.
  - **k-zero-day-safety** measures a number of zero-day vulnerabilities for an attacker to compromise a target.

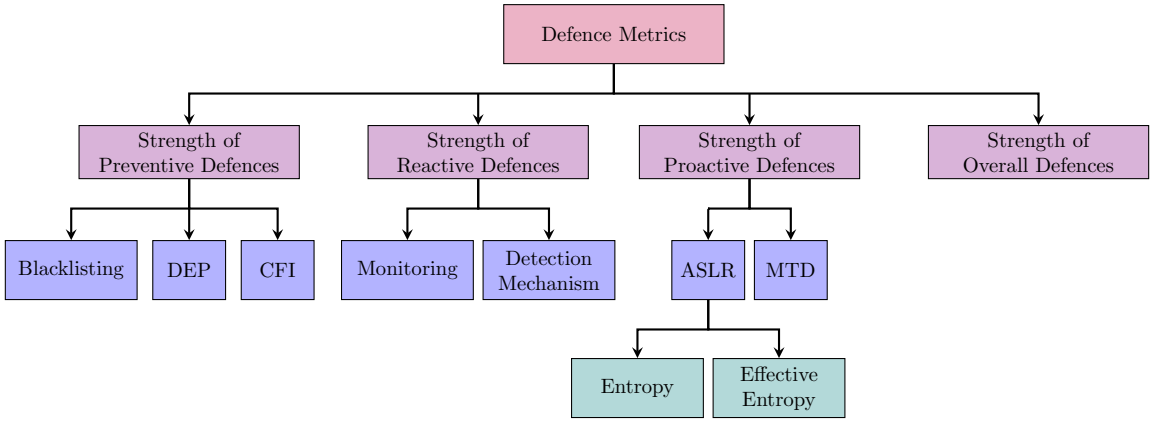
15.7.3.2 Probabilistic Severity Metrics

Deterministic metrics assume that vulnerabilities can certainly be exploited, implying that they measure what is possible rather than what is more likely than others. To be more realistic it is possible to associate a success probability to exploits. There are two approaches for defining probabilistic security metrics based on CVSS scores:

1. Metrics treating CVSS scores as atomic parameters (i.e. likelihood of exploitation metric).
2. Metrics not treating CVSS scores as atomic parameters.

15.8 Defence Metrics

Defence metrics aim to measure the strength of defence mechanisms placed in a system. We discuss how to measure the strength of preventive, reactive, and proactive defence mechanisms in addition to the strength of an overall system defence.



15.8.1 Metrics for Measuring the Strength of Preventive Defences

Preventive defences aim to block attacks. We focus on measuring the strength of the following preventive defences: blacklisting, Data Execution Prevention (DEP), and Control-Flow Integrity (CFI).

15.8.1.1 Metrics for Blacklisting

Blacklisting is a useful, lightweight defence mechanism. Suppose a malicious entity is observed at time t, then, traffic flowing to or from the malicious entity can be blocked starting at some time $t' \geq t$. Under this situation, two metrics can be derived:

- **Reaction time** metric captures the delay between the observation of the malicious entity at time t and the blacklisting of the malicious entity at time t' (i.e. $t' - t$), measured by an absolute scale.
- **Coverage** metric estimates the portion of blacklisted malicious players. This metric is measured by an ordinal scale.

15.8.1.2 Metrics for Data Execution Prevention (DEP)

No metrics have been defined to measure the effectiveness of DEP. The effectiveness of DEP can be measured based on the probability of being compromised by a certain attack $A(t)$ over all possible classes of attacks.

15.8.1.3 Metrics for Control-Flow Integrity (CFI)

We discuss three metrics to measure the quality of CFI mechanism:

- **Average indirect target reduction** measures the overall reduction in terms of the number of targets exploitable by the attacker where smaller targets are more secure. A defence leading to few but large targets offers less security than a defence leading to more, but smaller, targets. This metric can be measured by an absolute scale.
- **Average target size** is defined as the ratio between the size of the largest target and the number of targets. The smaller the ratio, the better the security.
- **Evasion resistance** is measured against control flow bending attacks, reflecting the effort (or premises) that an attacker must make (or satisfy) for evading the CFI scheme.

15.8.2 Metrics for Measuring the Strength of Reactive Defence

Detection mechanisms are well-known strategies for reactive defences, including intrusion detection systems (IDSs) and anti-malware programs.

15.8.2.1 Metrics for Monitoring

Attackers can be detected by monitoring mechanisms. Current monitoring practices only consider configurations of intrusion detection monitors, but not monitoring cost or impact of the existing compromised monitors. The common metrics to measure the quality of monitoring mechanisms are:

- **Coverage** metric measures the fraction of events detectable by a specific sensor deployment, reflecting a defender's need in monitoring events.
- **Redundancy** metric estimates the amount of evidence provided by a specific sensor deployment to detect an event. The amount of redundancy can be counted by the amount of sensors providing same information towards a same event.
- **Confidence** metric measures how well-deployed sensors detect an event in the presence of compromised sensors. This task needs to quantify truthfulness of the reports received from individual sensors.
- **Cost** metric measures the amount of resources consumed by deploying sensors including the cost for operating and maintaining sensors.

15.8.2.2 Metrics for Detection Mechanisms

Detection mechanisms can be measured via their individual, relative, and collective strengths.

- **Detection time**: For instrument-based attack detection, this metric is used to measure the delay between the time t_0 at which a compromised computer sends its first scan packet and the time t that a scan packet is observed by the instrument.
- **Intrusion detection metrics**: For IDS, including anomaly-based, host-based, and network-based IDS, their strength can be measured by:
  - **True-positive rate** is the probability that an intrusion is detected as an attack.
  - **False-negative rate** is the probability that an intrusion is not detected as an attack.
  - **True-negative rate** is the probability that a non-intrusion is not detected as an attack.
  - **False-positive rate** is the probability that a non-intrusion is detected as an attack.
  - **Intrusion detection capability metric** is the normalized metric.
  - **Receiver operating characteristic** (ROC) curve reflects the dependence of the true-positive rate on the false-positive rate, reflecting a trade off between the true-positive and the false-positive rates.
  - **Intrusion detection operating characteristic** (IDOC) curve describes the dependence of the true positive rate on the Bayesian detection rate, while accommodating the base rate.
  - **Cost metric** includes the damage cost incurred by undetected attacks, the response cost spent on the reaction to detected attacks including both true and false alarms, and the operational cost for running an IDS.
- **Relative Strength** metric reflects the strength of a defence tool when employed in addition to other defence tools. A defence tool does not offer any extra strength if it cannot detect any attack undetected by other defence tools in place. The relative strength of a defence tool $d' \in D'$ with respect to a set of defence tools $D \subset D'$ is defined as $\frac{|X_{d'} - U_{d \in D} X_d|}{|\mathcal{A}|}$.
- **Collective Strength** metric measures the collective strength of IDSs and anti-malware programs. The collective detection strength of defence tools is defined as $\frac{|U_{d \in D} X_d|}{|\mathcal{A}|}$.

In both of them, Relative and Collective Strength, $\mathcal{A}$ denotes a set of attacks, D denotes a set of defence tools and X_d denotes the set of attacks detected by a defence tool $d \in D$.

15.8.3 Metrics for Measuring the Strength of Proactive Defences

We discuss metrics for measuring two major proactive defence mechanisms, Address Space Layout Randomization (ASLR) and Moving Target Defence (MTD). These mechanisms are proactive because a system can be constantly re-configured to hinder the attack process. On the other hand, preventive and reactive defences often only incur changes to the defence tools (e.g. updating malware detection signatures or rules). Two metrics for measuring the strength of ASLR are:

- **Entropy** metric measures the entropy of a memory section, while noting that a greater entropy would mean a greater effort in order for an attacker to compromise the system.
- **Effective entropy** metric measures the entropy in a memory section that the attacker cannot circumvent by exploiting the interactions between memory sections.

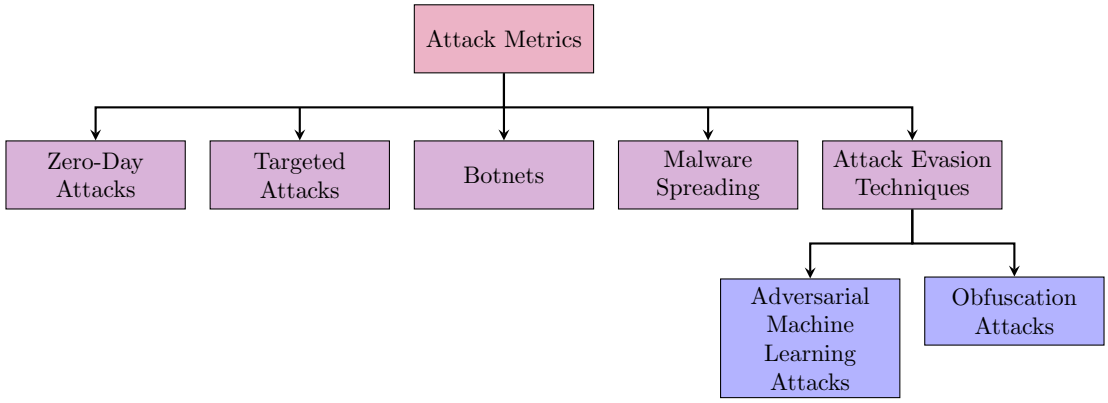
15.8.4 Metrics for Measuring the Strength of Overall Defence

We discuss the two metrics aiming to measure the strength of overall defences of a given system as follows:

- **Penetration resistance** (PR) can be measured by running a penetration test to estimate the level of effort (e.g. person-day or cost) required for a red team to penetrate into a system. This metric can be used to compare the defence strength of two systems against a same red team.
- **Network diversity** (ND) measures the least or average effort an attacker must make to compromise a target entity based on the causal relationships between resource types to be considered as the inclusion in an attack graph.

15.9 Attack Metrics

Attack metrics measure the strength of attacks performed against a system.



15.9.1 Measuring Zero-Day Attacks

Two metrics to measure how many zero-day attacks were launched during certain past period are:

- **Lifetime of zero-day attacks** measures the period of time between when an attack was launched and when the corresponding vulnerability is disclosed to the public.
- **Victims by zero-day attacks** measures the number of computers compromised by zero-day attacks.

15.9.2 Measuring Targeted Attacks

The success of targeted attacks or Advanced Persistent Threats often depends on the delivery of malware and the tactics to lure a target to open malicious email attachments. Let α denote a social engineering tactic, ranging from the least sophisticated to the most sophisticated (e.g. $\alpha \in 0, \ldots, 10$). Let β denote a technical sophistication of the malware in the attacks, ranging from the least sophisticated to the most sophisticated (e.g. $\beta \in [0, 1]$). The **targeted threat index** metric, indicating the level of targeted malware attacks, can be defined as $\alpha \cdot \beta$.

15.9.3 Measuring Botnets

The threat of botnets can be characterized by the following metrics:

- **Botnet size** refers to the number of bots, x, that can be instructed to launch attacks (e.g. Distributed Denial of Service attacks) at time t, denoted by $y(t)$.
- **Network bandwidth** indicates the network bandwidth that a botnet can use to launch Denial of Service attacks.
- **Botnet efficiency** can be defined as the network diameter of the botnet network topology. It measures a botnet's capability in communicating command-and-control messages and updating bot programs.
- **Botnet robustness** measures the robustness of botnets under random or intelligent disruptions.

15.9.4 Measuring Malware Spreading

Malware spreading is a common attack where a malware can spread out at a certain rate. The infection rate metric, denoted by γ, measures the average number of vulnerable computers that are infected by a compromised computer (per time unit) at the early stage of spreading.

15.9.5 Measuring Attack Evasion Techniques

Sophisticated attacks can evade defence mechanisms placed in a system using several strategies. We discuss two types of metrics for measuring the strength of those types of attacks: adversarial machine learning and obfuscation.

15.9.5.1 Measuring Adversarial Machine Learning Attacks

In adversarial machine learning, attackers can manipulate some features that are used in the detection models (e.g. classifiers). The spectrum of evasion scenarios includes that an attacker knows:

- a set of features used by a defender;
- both a set of features and training samples used by the defender;
- a set of features, the training samples, and the attack detection model (e.g. classifiers) used by the defender.

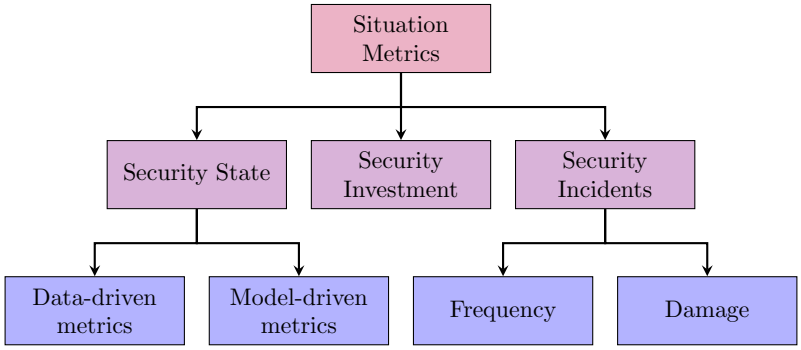
15.9.5.2 Measuring Obfuscation Attack

Obfuscation based on tools such as run-time packers have been widely used by malware writers to defeat static analysis. Little is understood about how to quantify the obfuscation capability of malware, except for the following two metrics:

- **Obfuscation prevalence** metric measures the occurrence of obfuscation in malware samples.
- **Structural complexity** metric measures the run-time complexity of packers in terms of their layers or granularity.

15.10 Situation Metrics

Situation metrics, denoted as `situation(t)`, reflect the comprehensive manifestation of attack-defence interactions with respect to an enterprise or computer system. These metrics can be divided into three sub-categories: security states $S(t)$, historical security incidents, and historical security investments.



15.10.1 Measuring Security State

We discuss the following two types of security state metrics in this section: data-driven and model-driven.

15.10.1.1 Data-Driven Metrics

This type of metric measures system state based on data. The examples of data-driven state metrics are:

- **Network maliciousness** metric estimates the fraction of blacklisted IP addresses in a network.
- **Rogue network** metric captures the population of networks used to launch drive-by download or phishing attacks.
- **ISP badness** metric quantifies the effect of spam from one ISP or Autonomous System (AS) on the rest of the Internet.
- **Control-plane reputation** metric calibrates the maliciousness of attacker-owned (i.e. rather than legitimate but mismanaged/abused) ASs based on their control plane information.
- **Cybersecurity posture** metric measures the dynamic threat imposed by the attacking computers.

15.10.1.2 Model-Driven Metrics

This type of metric measures system states in terms of the outcome of attack-defence interaction models. The two most used metrics are:

- Fraction of compromised computers.
- Probability a computer is compromised at time t.

15.10.2 Measuring Security Incidents

Measuring security incidents is another aspect of `situation(t)`. To obtain the severity and impact of incurred security incidents, we discuss how to measure the frequency and damage of the security incidents.

15.10.2.1 Measuring Frequency of Security Incident

Frequency of security incidents can be measured by:

- **Encounter rate** measures the fraction of computers that encountered some malware or attack during a period of time.

- **Incident rate** measures the fraction of computers successfully infected or attacked at least once during a period of time.
- **Blocking rate** is the rate an encountered attack is successfully defended by a deployed defence.
- **Breach frequency** metric measures how often breaches occur.
- **Breach size** metric gives the number of records breached in individual breaches.
- **Time-between-incidents** metric measures the period of time between two incidents.
- **Time-to-first-compromise** metric estimates the duration of time between when a computer starts to run and the first malware alarm is triggered on the computer where the alarm indicates detection rather than infection.

15.10.2.2 Measuring Damage of Security Incident

The damage to a system after a security incident occurs can be estimated based on the degree of impact caused by the security incident. The impact can be estimated by the following metrics:

- **Delay in incident detection** measures the time between the occurrence and detection, implying that a longer delay is a higher damage.
- **Cost of incidents** may include both the direct cost (e.g. the amount of lost money) and the indirect cost (e.g. negative publicity and/or the recovery cost).

15.10.3 Measuring Security Investment

Investment to ensure an enterprise's security can be measured as follows:

- **Security spending** indicates a percentage of IT budget. This metric is important because enterprises want to know whether their security expenditure is justified by the security performance and is comparable to other organizations' security investments.
- **Security budget allocation** estimates how the security budget is allocated to various security activities and resources.
- **Return on security investment** (ROSI) is a variant of the classic return on investment (ROI) metric, measuring the financial net gain of an investment based on the gain from investment minus the cost of investment. Since security is not a real investment (not generating a revenue), the ROSI metric actually measures the reduction in the loss caused by incompetent security.
- **Net present value** measures the difference between the present economic value of future inflows and the present economic value of outflows with respect to an investment.

16 Case Study: The PANOPTESEC System

The PANOPTESEC Consortium deliver a beyond-state-of-the-art prototype of a cyber defence decision support system, demonstrating a risk-based approach to automated cyber defence that accounts for the dynamic nature of Information and Communications Technologies (ICT) and the constantly evolving capabilities of cyber attackers.

16.1 MAPE-K cycle

The basic idea is that the system is semi-automated, meaning that the final mitigation action must be validated and performed by a human analyst, but the rest of the system is fully automated (from information gathering, passing for the analysis up to the elaboration of the mitigation action).

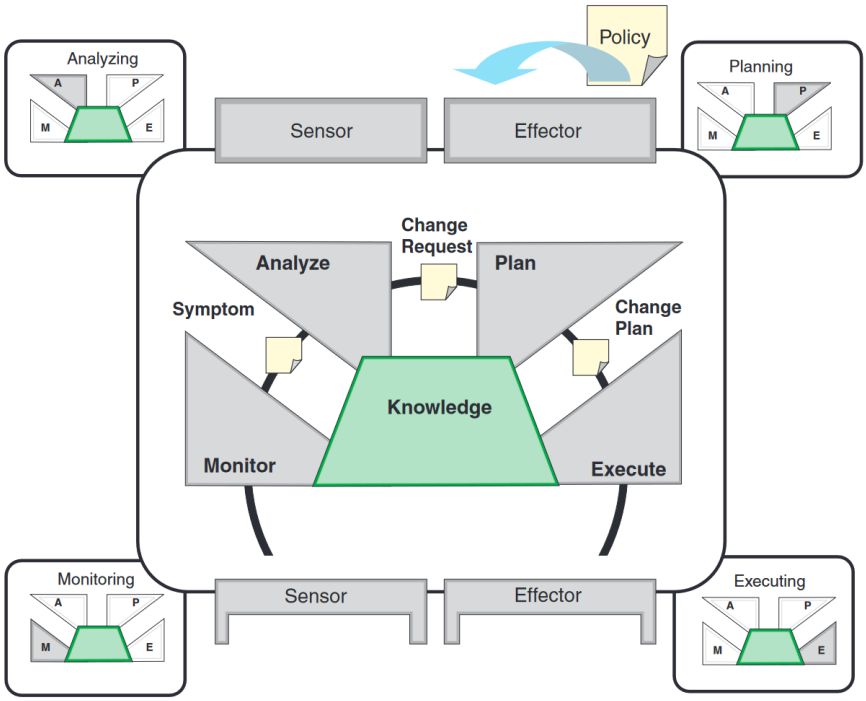


Figure 48: Functional details of the autonomic manager from [26].

The purpose of the tool is, as mentioned, support decision for cyber vulnerability, incident detection and response management. This can be achieved with the 4 phases of the MAPE-K cycle:

- The **monitor** function provides the mechanisms that collect, aggregate, filter and report details (such as metrics and topologies) collected from a managed resource.
- The **analyze** function provides the mechanisms that correlate and model complex situations. These mechanisms allow the autonomic manager to learn about the IT environment and help predict future situations.
- The **plan** function provides the mechanisms that construct the actions needed to achieve goals and objectives. The planning mechanism uses policy information to guide its work.
- The **execute** function provides the mechanisms that control the execution of a plan with considerations for dynamic updates.

These four parts work together to provide the control loop functionality. Figure 48 shows a structural arrangement of the parts rather than a control flow. The four parts communicate and collaborate with one another and exchange appropriate knowledge and data. Autonomic managers provide sensor and effector manageability interfaces for other autonomic managers and manual managers to use. Using standard sensor and effector interfaces enables these components to be composed together in a manner that is transparent to the managed resources.

16.2 Architecture

The goal of this task is to develop the system high-level design including identification and selection of underlying software architectural technologies, applicable standards, and interface specifications for the embedded software components comprising the PANOPTESEC system. This task will also involve the specification of the data model required to support the embedded software components of the PANOPTESEC system. The data collection system, which also contains a model for describing the impact of cyber-attacks as well as corresponding countermeasures, will provide the necessary input for other components of the PANOPTESEC system. Figure 49 illustrates the global architecture and logical data flows in PANOPTESEC, and puts in evidence the central role of the Data Collection and Correlation System.

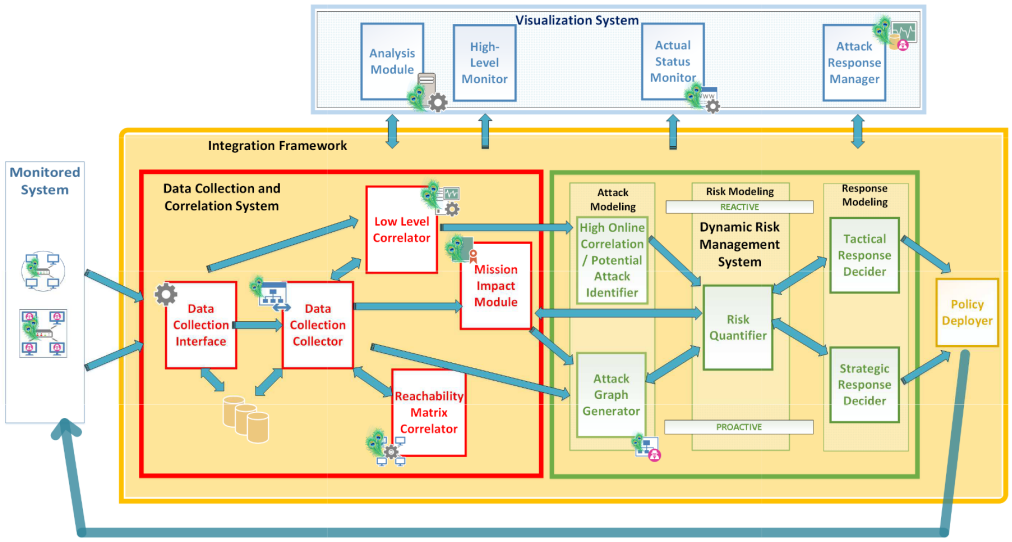


Figure 49: Data collection and correlation as part of the PANOPTESEC global architecture and logical data flow from [27].

The Data Collection Interface (DCI) is responsible to set up connections to monitored systems, such that the Data Collection Collector component gets access to data produced by monitored systems. The Data Collection Collector (DCC) is in charge of accessing and pre-processing data retrieved from resources provided by monitored systems. The idea of the Low Level Correlator (LLC) is to provide a real-time stream of events and alerts for other modules.

16.3 General Approach

16.3.1 Continuous Proactive chain

Proactive response system is a strategic response. Providing information for something that will be implemented over the long term and could provide mitigations that take some time to implement. The idea is to focus on an attack path that could exist in the system and that has high priority in terms of impact on assets.

- **Monitor**: Collect network and security relevant information from diverse data sources and build the following correlated information:
  - Network Inventory
  - Vulnerability Inventory
  - Mission Graph (identifies critical supporting assets)
  - Network and system dependency model (from real-time flow data)
  - Reachability matrix
- **Analyze**: Perform security analysis of collected information to:
  - Generate the Attack Graph from hypothetical source to critical supporting assets
  - Quantify risk to critical supporting assets

- **Plan**: Conduct automated decision support analysis to:
  - Identify potential response plans to reduce risk
  - Evaluate response plans against business/mission and financial impact
  - Propose prioritized response plans
- **Execute**: Prepare and issue selected response plans:
  - Response plans may consist of several mitigation actions
  - Defined according to acceptable policies
  - Formatted for connected deployment capability
- **Knowledge Base**: Provides access to relevant data at different levels of detail/abstraction:
  - Contains raw data collected by the system
  - Contains current and historical results of analytic processes
- **Sensors, Effectors & Scanners**: Out of scope. Leverage third party products, in fact, the System can adapt to many different topology/inventory/vulnerability scanners or different data sources.

16.3.2 Continuous Reactive chain

Reactive response system is a tactical response performed at run time. We always use an attack graph as a model to identify attack patterns, but instead of focusing on the one with the greatest impact on the system, we focus on those that can potentially be followed by an attacker at that moment (deduction through alerts).

- **Monitor**: Augment proactive data with real-time (reactive) incident data:
  - Network events
  - Intrusion events
- **Analyze**: Perform security analysis of collected information to:
  - Localize incidents on Attack Graphs
  - Quantify risk to critical supporting assets
- **Plan**: Conduct automated decision support analysis to:
  - Identify potential response plans to reduce risk
  - Evaluate response plans against business/mission and financial impact
  - Propose prioritized response plans
- **Execute**: Prepare and issue selected response plans:
  - Response plans may consist of several mitigation actions
  - Defined according to acceptable policies
  - Formatted for connected deployment capability
- **Knowledge Base**: Provides access to relevant data at different levels of detail/abstraction:
  - Contains raw data collected by the system
  - Contains current and historical results of analytic processes
- **Sensors & Effectors**: Out of scope. Leverage third party products. The System can adapt to many different sensors and firewalls.

16.3.3 Modularized architecture

OSGi based integration framework Component Composition and Service Integration layers deliver a modular architecture Loosely coupled modules support diverse options deployment as both self contained system and distributed environments.

16.3.4 Simulation Environment

The Simulation Environment (Sim-Env) has been created starting from ACEA Distribution Energy environment (Rome), using the resources of the Disaster Recovery site. The Disaster Recovery systems used are real operational systems in "cold standby" mode. These are then augmented by real (standby and test) equipment with virtual clones in order to "emulate" the scale of the operational environment. The Sim-Env for PANOPTESEC Project is composed of several logical blocks, listed below:

- Emulation Environment
- Developing Environment
- Partners Portal for PANOPTESEC Project Development and Testing
- PANOPTESEC Demo Environment

16.4 Data Flow: Proactive View

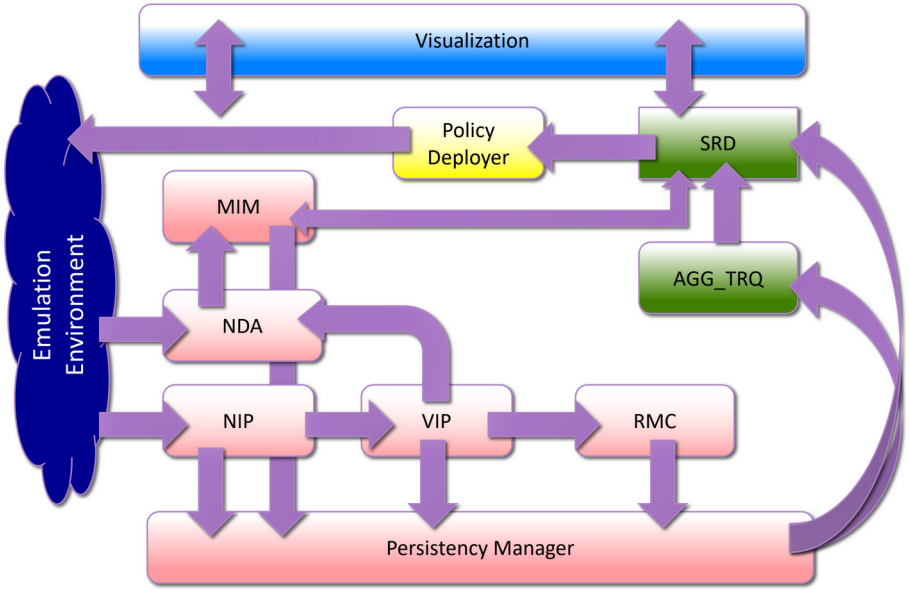


Figure 50: The PANOPTESEC Data Flow: Proactive View.

Pink modules represent data collection and pre-processing of correlation data delivered for analysis, Network Inventory Processor (NIP), Vulnerability Inventory Processor (VIP), Reachability Matrix Correlator (RMC) are stored in the Persistency Manager. The MIM (Mission Impact Module) and NDA (Network Dependency Analyzer) modules are used to support the business section of the risk analysis. In green, we have AGG_TRQ, where we have the graph analysis (Attack Graph Generator) and we estimate the risk (Threat Risk Quantifier), and the SRD (Strategic Response Decider) which is responsible for planning the mitigation actions. The yellow module concerns the development, finally in blue there is the visual analytical environment. Note that the loop in Figure 50 is synchronous to avoid to make analysis on a snap shoot inconsistent. This proactive chain is a Dynamic Risk Management Response System (DRMRS) that has a modular architecture composed of several integrated elements that interact to evaluate, assess and mitigate the impact of identified threats. Components are grouped according to their functions in three main blocks as shown in Figure 51:

- The **input data** that provide the information required to analyze the impact of threats and mitigation actions on the target system.
- the **processing modules** composed of the Strategic Response Decider (SRD), the Attack Graph Generator (AGG), the Response Operational Impact Assessment (ROIA), and the Threat Risk Quantifier (TRQ), whose mission is to evaluate individual and combined mitigation actions in financial and operational perspectives in order to generate the corresponding response plans.

- The **output data**, composed of the primitive and/or enriched response plans, which indicate the security actions to be deployed in order to mitigate the current threats.

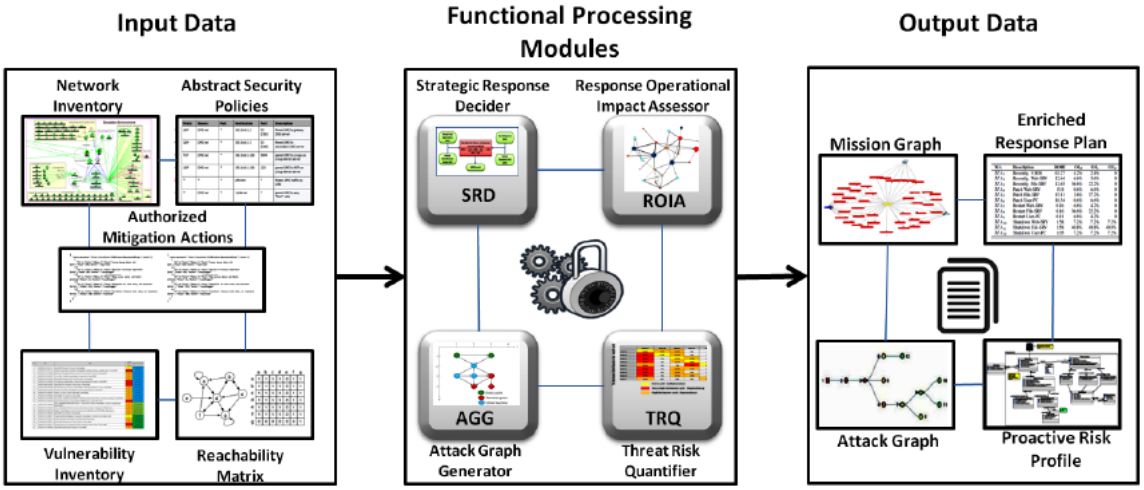


Figure 51: Dynamic Risk Management Response System (DRMRDS) conceptual workflow on acquiring, processing, and generating data from [28].

16.4.1 Input Data

Now let's see how the input data will be translated through the functional modules into output data.

- The **Network Inventory** contains information of active devices of the monitored system. Each device is marked with a unique identifier (ID), a given type (PEP_Type), annual cost (AEC) and associated vulnerabilities (CVEs). The sum of the AEC for all devices result into the annual infrastructure value (AIV), used to compute the RORI index.
- **Abstract security policies** include information required for the instantiation of security policies: threats, policy enforcement points (PEP) and mitigation actions. Each threat defined in an abstract default security policy is also described by a threat ID, severity, frequency and likelihood threshold.
- **Authorized mitigation actions** includes a list of mitigation actions that are authorized to be executed as a response to a given threat. Each mitigation action is defined by an identifier, a scope of enforcement, indicating whether it is applicable for strategic or tactical response; the enforcement points on which it is permitted to be deployed; the annual equipment value associated to the Policy Enforcement Point (PEP) and the annual response cost associated to the enforcement of the action.

16.4.2 Functional Processing Modules

The functional modules responsible for the generation of the response plans are:

- The Attack Graph Generation (AGG), whose purpose is to calculate the exposure of the monitored system to threats.
- The Threat Risk Quantification (TRQ), that aims at assessing the risk of each detrimental events feared by the organization, that uses the monitored system to support its business processes.
- The Response Operational Impact Assessor (ROIA), whose objective is to assess potentials of collateral damage onto a company and their associated business processes.
- The Strategic Response Decider (SRD), that analyses response plans composed of actions that mitigate the current situation.

16.4.2.1 Attack Graph Generation (AGG)

According to characteristics commonly used in the literature to classify attack graphs those generated are topologically directed graphs. This means each directed edge represents the attack action from the tail node towards the head node. Each node corresponds to the compromising status of a specific device of the monitored system. To generate an attack graph, the AGG bases its computation on three kinds of inputs: reachability between nodes, the vulnerability on nodes and privilege level gained on nodes. We will have a logical inference problem transformed into a graph theory problem. The **innovation** parts are that we have non-monotonic attack paths i.e. backtracking, on already compromised nodes to escalate the compromising status from an unprivileged User level to a privilege Root level and generation optimizations (more than 100x) to cluster Attack Paths in case of full-mesh networks. The **downside** is that in fully-meshed networks we have exponential growth of Attack Paths number which is not manageable for High-Level Online alert Correlators.

16.4.2.2 Threat Risk Quantification (TRQ)

The TRQ functional module implements a feature that assesses a risk weighing on an Organization. TRQ exploits the detrimental events, and their link with the business resource through the business model of the mission graph, to compute elementary risks (ER) based on computed attack paths. To calculate Likelihood by considering an attack path, which is basically a sequence of exploited vulnerabilities, as an equivalent Markov chain in which the k^{th} state T_k of the Markov chain corresponds to the k^{th} step in the attack path. In the equivalent Markov chain, we set the exit rate λ_k of the sojourn time of the k^{th} state T_k to a value that is homogeneous to the difficulty score at the k^{th} step in the attack path. The difficulty score is computed based on parameters of metrics of the Common Vulnerability Scoring System version 2 (CVSSv2), extracted from the National Vulnerability Database (NVD), as presented in Equation 1:

$$\lambda_k = RoundTo1(AV \cdot AC \cdot Auth \cdot Expl \cdot RC) \tag{1}$$

From Equation 1, the function suggests to round up to one the product of the Access Vector (AV), Access Complexity (AC), Authentication (Auth), Exploitability (Expl) and Report Confidence (RC) CVSSv2 metrics of the related exploited vulnerability at step k of the corresponding attack path. The Mean Time to Attack Object (MTAO) is first computed, comparably to the Mean Time To Failure (MTTF) in the dependable theory, as the summation of the expectation of the mean sojourn time of each state in the equivalent Markov chain of the considered attack path (see Equation 2).

$$MTAO = \sum_k E\{T_k\} = \sum_k \frac{1}{\lambda_k} \tag{2}$$

The likelihood of the attack path associated to ER_i is calculated as proposed in Equation 3.

$$Likelihood_i = -20 \cdot log_{10}(\frac{MTAO - MTAO_{min}}{MTAO}) \tag{3}$$

What we can observe from the likelihood is that the central part of the metric, the normalization of MTAO, is always in the interval $[0, 1]$. Furthermore, the metric grows in a non-linear manner with the progress of the attacker in the path. For two attack paths with the same number of steps, the lowest likelihood value is assigned to the attack path with the easiest vulnerability exploitation. From the paper [28], $MTAO_{min}$ is the lowest possible value for an $MTAO$. It corresponds to the simplest attack path, composed of one attack step exploiting the easiest possible CVSS and we can substitute the value $MTAO_{min}$ with 1. While the attack scenario (attack path) is relevant to calculate the likelihood, the impact of each ER_i (i.e. the Elementary Impact denoted EI_i) is decided by what follows the successful execution of the terminal step of an attack scenario. The consequences of V can be expressed with the Boolean variables that indicate whether the successful exploitation of V leads to a violation of confidentiality (Imp_C), integrity (Imp_I), or availability (Imp_A). The detrimental event associated to EI_i has two main properties that are obtained during the process of a traditional risk assessment:

- $Viol_C$, $Viol_I$, and $Viol_A$ are Boolean variables that indicate whether the detrimental event arises following a violation of confidentiality, integrity and availability respectively.
- Magnitude is a quantitative or qualitative variable that indicates the impact's magnitude on the organization.

EI_i is calculated by crossing the technical consequences of the attack scenario (Imp_C, Imp_I and Imp_A) with the nature of the Detrimental Event ($Viol_C$, $Viol_I$ and $Viol_A$) as presented in Equation 4.

$$EI_i = [(Imp_C \wedge Viol_C) \vee (Imp_I \wedge Viol_I) \vee (Imp_A \wedge Viol_A)] \cdot Magnitude \quad (4)$$

Each possible attack path computed on the proactive perspective may lead to zero, one or several detrimental events DE_j. To a given possible attack path related to DE_j, denoted P, correspond a set of proactive elementary impacts $\{PEI_{ij}^{p_j=P}\}$ (i.e. elementary impacts on DE_j caused by P), and a set of proactive elementary risks $\{PER_{ij}^{p_j=P}\}$ (i.e. elementary risks relative to DE_j caused by P) expressed in terms of Likelihood of the attack path P and impact its $PEIs$. The proactive risk profile is composed by aggregating the various computed elementary risks using Equation 5.

$$Risk_{DE_j} = max(PER_{ij}^{p_j}) \quad (5)$$

16.4.2.3 Response Operational Impact Assessor (ROIA)

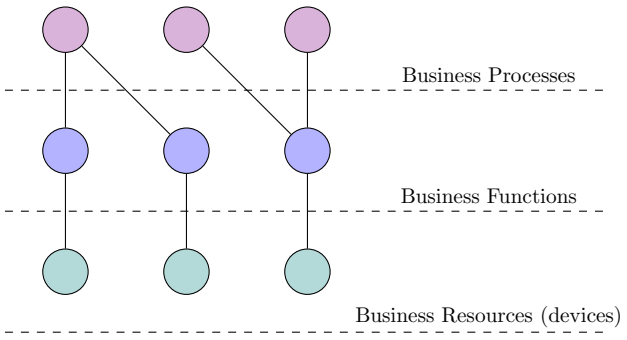
The Response Operational Impact Assessor (ROIA) aims at using a mission impact assessment to ponder about negative side-effects of response plans and individual mitigation actions. Any executed action to prevent a proactive infiltration of a network and individual nodes, such as patching nodes or strategically enforced firewall rules, or any reactive measure to mitigate an ongoing attack, such as ad-hoc connection drop-downs or deactivation of individual nodes, inherently represents a threat to the operational capability on involved nodes. For instance, shutting down a node in a network will inevitably reduce the operational capability of this node with a probability of one. Further, employing a patch on a node might lead to an immediate conflict with a probability of 10%, requires a reboot in some intermediate time. We call this reduction of operational capability, an impact on a node. Moreover, what needs to be considered is that any local impact may spread throughout a network: if a node is highly dependent on receiving information from a node which has been shutdown, it definitely will be impacted as well as it is not able to operate as intended anymore.

Resource Dependency Model

A resource dependency model is a mathematical model representing dependencies of involved resources supporting a company in the form of a Probabilistic Graphical Model (PGM). The PGM represents every involved resource as a random variable and represents every dependency among them as a conditional probability of failure. A resource dependency model is constantly and automatically learned from network traffic analyses. By doing so, it supports a mission dependency model in identifying mission critical devices and dynamically captures changing environments.

Mission Dependency Model

A mission dependency model is a mathematical model representing dependencies among a company or mission in the form of a Probabilistic Graphical Model (PGM) that captures dependencies of a company or mission on its business processes that need to be accomplished. Each node of a mission dependency model represents a random variable, and every dependency represents a conditional probability of impact.



16.4.2.4 Strategic Response Decider (SRD)

The Strategic Response Decider (SRD) handles identified threats, authorized mitigation actions and strategic policies. It extracts concrete entities from reported threats, and infers concrete policy instances to eventually guide the system into new updates and re-configurations. These are provided as concrete response plans on a long-term proactive perspective. The goal of the SRD is the automated administration of policy-related activities, including addition of new rules, removal of unnecessary conditions, and activation of strategic responses. The SRD relies on the Response

Financial Impact Assessor (RFIA) component to quantify the level of benefit perceived per response plan on a financial basis. The RFIA provides an assessment concerning the potential financial impact that a given response plan may cause to an organization. Response plans represent proposed mitigation of the assessed risks and are assumed to be composed of one or more mitigation actions. The RFIA performs the calculation of the return-on-response-investment (RORI) index associated to the mitigation actions composing a response plan. The RORI index is used to evaluate optimal plans, by ranking them as a trade-off between their efficiency in stopping potential attacks, and their ability to preserve, at the same time, the best service to legitimate users. The RORI index is calculated for each mitigation action, according to Equation 6.

$$RORI = \frac{(ALE \cdot RM) - ARC}{ARC + AIV} \cdot 100 \tag{6}$$

Where ALE (annual loss expectancy) refers to the financial cost expected from the threat, in the absence of applying mitigation; RM (risk mitigation) estimates the effectiveness and coverage of an action in mitigating the threat; ARC (annual response cost) expresses the expected cost of applying the mitigation action; and AIV (annual infrastructure value) is a fixed cost associated to the system infrastructure, regardless of applying or not mitigation.

16.4.3 Output Data

- The **Mission Graph** describes the business model of an organization. It associates to each asset one or several business processes and any failure of these processes will lead to an impact on the company, mission or organization.
- A **Proactive risk profile** contains structured information representing the risk posture of the monitored system on the mid-long term.
- An **enriched response plan** contains detailed information about the actions that best mitigate the threat scenarios.

16.5 Data Flow: Reactive View

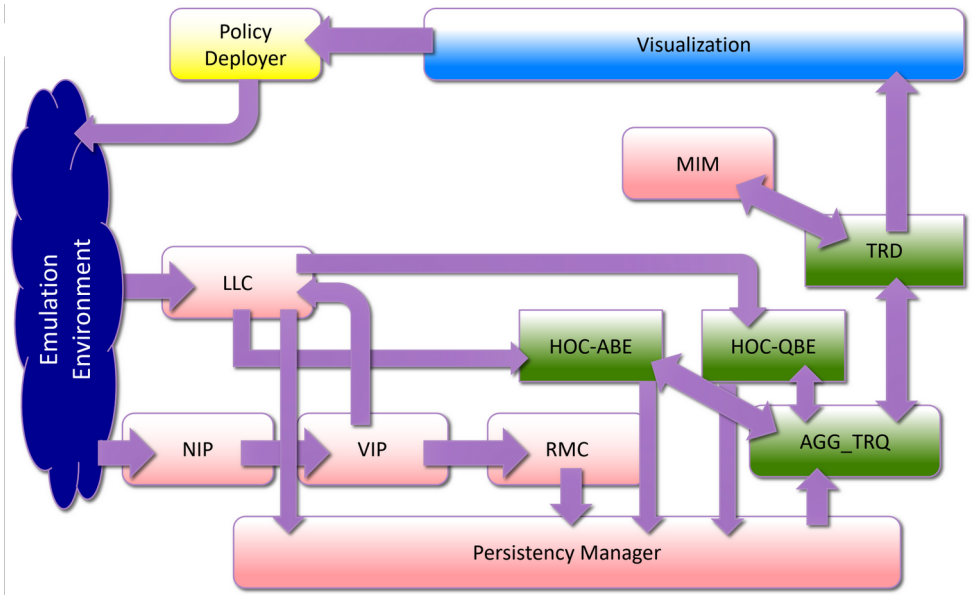


Figure 52: The PANOPTESEC Data Flow: Reactive View.

Note that the loop in Figure 52 is asynchronous because we are in an emergency state so we accept that there are some inconsistencies in the analyzed snapshot.

16.5.1 On-line Multi-step Attack Detector

The first and fundamental feature of the On-Line Correlation Engine is the ability to understand if an alert matches with an edge of one or more Attack Paths (AP) in the attack graph. If this

happens, the alert must be considered to estimate which is the more probable AP followed by the attacker among all those with a matching edge [29].

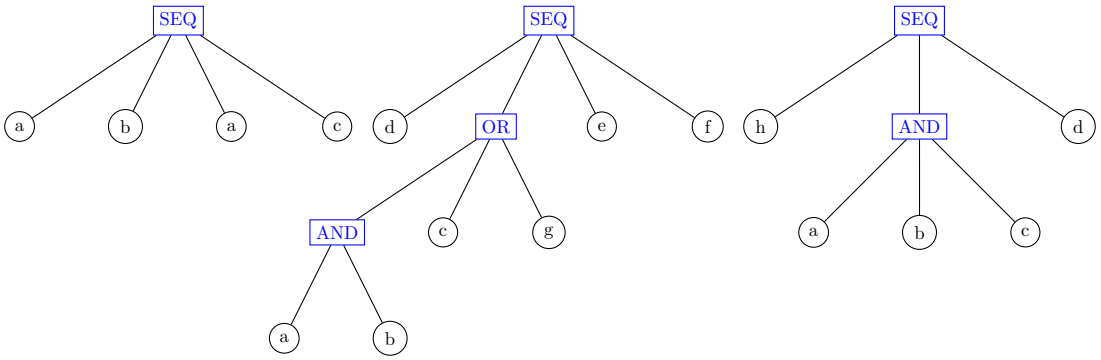


Figure 53: Correlation trees of Attack Graphs.

In order to enable the system to process a huge amount of paths, each attack graph is transformed into 3 possible structures called correlation queries. These are shown in Figure 53, in fact, the graph is used to compute spanning attack trees but only scenarios that do not exceed 4 to 5 steps are considered. In practice, an attacker prefers to follow a short path. Furthermore a longer path often contains all the attack steps of a shortest one already identified. We suppose also that the attacker is coming from outside the supervised system. These realistic assumptions significantly reduce the number of attack paths. The trees obtained are in fact attack trees where each attacker's action corresponds to a vulnerability exploited to perform the attack step [30].

16.5.1.1 QBE - Query Based Engine

We study the problem of having an attacker who enters the network through an entry point and begins to compromise the network nodes one by one by exploiting their vulnerabilities. The attack model we have here does not involve an offensive that ends in one step, the attacker does not focus on a single target, so we have to look at the problem from a broader point of view: to see if the set of exploits that could trigger an alert in the IDS may be compatible with a path in the graph. This translates into observing the alert sequence and identifying the path, when we see that the attacker is following a path in the graph, launch an alert. The challenges from this approach were handle the 3V (Volume, Variety, Velocity) and have the incomplete or inaccurate information (IDSs accuracy, unknown attacks).
To identify path attacks we rely on a detection technique based on the spatial and temporal correlation between events collected from different hosts. An innovative method because we don't focus on a single attack pattern and therefore we don't need a reference signature. However, we often find ourselves in situations where the information we are using is not accurate or complete. The basic idea of the mechanism is, for each attack path, to identify a query that is executed along the entire stream of alerts and that analyzes their content by trying to match them with an edge of the path, even if it does not start exactly from the first node, but from a subsequent one. Once the pattern has been identified, even partial, it will be entered in a match pattern, however, we cannot have a complete list of patterns, so we will also have an unmatched pattern that can be used to discover new attack patterns that have not yet been catalogued in a known match.

16.5.1.2 ABE - Automaton Based Engine

The Automaton Based Engine (ABE) must be able to take as input automatically generated correlation rules; predict incoming multi-step attacks; detect attack scenarios with some missing steps and scalable to handle hundreds of alerts per second while supporting thousands of correlation rules. The idea to achieve the objectives just listed is the following: taking as input a structure of the graph that we have to monitor, we can translate each correlation rule into automata, therefore encode the path as a finite state machine and interpret the alert stream as symbols that the automata take as input. We navigate the automata until we reach an accept state, which leads to the generation of an alert, or rejects, which leads to the rejection of the path.

References

[1] IT Governance Institute. *Board Briefing for IT Governance, 2nd Edition*. Information Systems Audit and Control Association, 2003. ISBN: 9781893209640. URL: https://books.google.it/books?id=MpD2PAAACAAJ.

[2] Rossouw Von Solms and S. H. (Basie) Von Solms. "Information Security Governance: A Model Based on the Direct-Control Cycle". In: *Comput. Secur.* 25.6 (Sept. 2006), pp. 408–412. ISSN: 0167-4048. DOI: 10.1016/j.cose.2006.07.005. URL: https://doi.org/10.1016/j.cose.2006.07.005.

[3] S.H. Solms and Rossouw Solms. "Information Security and Information Security Governance". In: Oct. 2009, pp. 1–11. ISBN: 978-0-387-79983-4. DOI: 10.1007/978-0-387-79984-1_3.

[4] ISO/TC 262. *Risk management – Guidelines*. Standard ISO 31000:2018, 2018. Geneva, Switzerland: International Organization for Standardization, 2018. URL: https://www.iso.org/obp/ui/#iso:std:iso:31000:ed-2:v1:en.

[5] U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA). *Common attack pattern enumeration and classification*. URL: https://capec.mitre.org/.

[6] U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA). *Common Weakness Enumeration*. URL: https://cwe.mitre.org/.

[7] Open Web Application Security Project OWASP. *OWASP Risk Assessment Framework*. URL: https://owasp.org/www-project-risk-assessment-framework/.

[8] SANS Institute. *A Qualitative Risk Analysis and Management Tool - CRAMM*. URL: https://www.sans.org/white-papers/83/.

[9] Club De La Securite De L'information Français. *MEHARI 2010 Fundamental concepts and functional specifications*. URL: http://meharipedia.x10host.com/wp/wp-content/uploads/2016/12/MEHARI-2010-Principles-Specifications.pdf.

[10] Open Web Application Security Project OWASP. *OWASP Risk Rating Methodology*. URL: https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology.

[11] Kerem Kaynar. "A taxonomy for attack graph generation and usage in network security". In: *J. Inf. Secur. Appl.* 29 (2016), pp. 27–56. DOI: 10.1016/j.jisa.2016.02.001. URL: https://doi.org/10.1016/j.jisa.2016.02.001.

[12] Kyle Ingols, Richard Lippmann, and Keith Piwowarski. "Practical Attack Graph Generation for Network Defense". In: *Proceedings of the 22nd Annual Computer Security Applications Conference*. ACSAC '06. USA: IEEE Computer Society, 2006, pp. 121–130. ISBN: 0769527167. DOI: 10.1109/ACSAC.2006.39. URL: https://doi.org/10.1109/ACSAC.2006.39.

[13] Vipin Kumar, Jaideep Srivastava, and Aleksandar Lazarevic. *Managing Cyber Threats: Issues, Approaches, and Challenges (Massive Computing)*. Berlin, Heidelberg: Springer-Verlag, 2005. ISBN: 0387242260.

[14] Karen A. Scarfone and Peter M. Mell. *SP 800-94. Guide to Intrusion Detection and Prevention Systems (IDPS)*. Tech. rep. Gaithersburg, MD, USA: National Institute of Standards & Technology, 2007. URL: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-94.pdf.

[15] Cachin C. Adelsbach A. et al. *MAFTIA Conceptual Model and Architecture*. Tech. rep. Department of Informatics, University of Lisbon, Nov. 2001. URL: http://hdl.handle.net/10451/14152.

[16] D. Moore et al. *The Spread of the Sapphire/Slammer Worm*. Tech. rep. CAIDA, ICSI, Silicon Defense, UC Berkeley EECS and UC San Diego CSE, Jan. 2003.

[17] Thomas Millar Paul Cichonski and Karen Scarfone Tim Grance. *SP 800-61. Computer Security - Incident Handling Guide Recommendations of the National Institute of Standards and Technology*. Tech. rep. National Institute of Standards & Technology, Aug. 2012. URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf.

[18] Pierre Jacobs, Alapan Arnab, and Barry Irwin. "Classification of Security Operation Centers". In: *2013 Information Security for South Africa*. 2013, pp. 1–7. DOI: 10.1109/ISSA.2013.6641054.

[19] Alissa Torres. *Building a World-Class Security Operations Center: A Roadmap*. Tech. rep. SANS Institute, May 2015.

[20] European Union Agency for Cybersecurity. *Good Practice Guide for Incident Management*. Tech. rep. European Union Agency for Cybersecurity (ENISA), Dec. 2010. URL: https://www.enisa.europa.eu/publications/good-practice-guide-for-incident-management.

[21] *Security and Privacy Controls for Information Systems and Organizations*. Sept. 2020. DOI: 10.6028/nist.sp.800-53r5. URL: http://dx.doi.org/10.6028/NIST.SP.800-53r5.

[22] *Risk management framework for information systems and organizations:* Dec. 2018. DOI: 10.6028/nist.sp.800-37r2. URL: http://dx.doi.org/10.6028/NIST.SP.800-37r2.

[23] Marcus Pendleton et al. "A Survey on Systems Security Metrics". In: *ACM Comput. Surv.* 49.4 (Dec. 2016). ISSN: 0360-0300. DOI: 10.1145/3005714. URL: https://doi.org/10.1145/3005714.

[24] Inc. (FIRST) FIRST.Org. *Common Vulnerability Scoring System version 3.1*. URL: https://www.first.org/cvss/v3.1/specification-document.

[25] U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA). *Common Weakness Scoring System (CWSS)*. Sept. 2014. URL: https://cwe.mitre.org/cwss/cwss_v1.0.1.html.

[26] *An Architectural Blueprint for Autonomic Computing*. Tech. rep. International Business Machines Corporation, June 2005. URL: https://www-03.ibm.com/autonomic/pdfs/AC%5C%20Blueprint%5C%20White%5C%20Paper%5C%20V7.pdf.

[27] PANOPTESEC Consortium. *Data Collection and Correlation Requirements*. Mar. 2015. URL: https://www.panoptesec.eu/dissemination/FP7-ICT-610416-PANOPTESEC_D411_v2.0-QA-Approved.pdf.

[28] G. Gonzalez-Granadillo et al. "Dynamic risk management response system to handle cyber threats". In: *Future Generation Computer Systems* 83 (2018), pp. 535–552. ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2017.05.043. URL: https://www.sciencedirect.com/science/article/pii/S0167739X17311433.

[29] Marco Angelini et al. "An Attack Graph-Based On-Line Multi-Step Attack Detector". In: *Proceedings of the 19th International Conference on Distributed Computing and Networking*. ICDCN '18. Varanasi, India: Association for Computing Machinery, 2018. ISBN: 9781450363723. DOI: 10.1145/3154273.3154311. URL: https://doi.org/10.1145/3154273.3154311.

[30] David Lanoe, Michel Hurfin, and Eric Totel. "A Scalable and Efficient Correlation Engine to Detect Multi-Step Attacks in Distributed Systems". In: *2018 IEEE 37th Symposium on Reliable Distributed Systems (SRDS)*. 2018, pp. 31–40. DOI: 10.1109/SRDS.2018.00014.