Enhancing the Security of High-Responsibility Information Systems Through Fault Tree Modeling

Constantin-Alin COPACI, Ioan C. BACIVAROV
Faculty of Electronics, Telecommunications and Information Technology,
National University of Science and Technology POLITEHNICA Bucharest, Romania
constantin.copaci@stud.etti.upb.ro, ioan.bacivarov@upb.ro

Abstract

With the advancement of technology, high-dependability information systems have become indispensable for the efficient operation of activities in strategic sectors, among which communications play a crucial role. This paper aims to ensure the security and reliability of a communication system through the application of fault tree analysis. This approach seeks to identify vulnerable components, assess the probability of adverse events within the system, and propose measures to enhance its performance under conditions of risk or failure.

Index terms: communication system, failure, Fault Tree, reliability, security

1. Introduction

High-dependability information systems (HDIS) are technical or computer-based systems whose correct, continuous, and secure operation is essential for preventing serious consequences that could affect human life, the environment, public safety, or critical infrastructure.

Ensuring security, reliability, and fault tolerance constitutes a fundamental requirement for high-dependability information systems, as any malfunction or incident may have significant repercussions. In this context, Fault Tree Analysis (FTA) serves as a key tool, as it enables the logical and structured modeling and assessment of potential causes leading to system failures [1], [2].

2. Fault Tree modeling for high-dependability information systems

Fault Tree modeling represents a fundamental stage in the reliability and security analysis of high-dependability information systems, as it enables the identification and assessment of factors that may lead to failures or undesired events. This section presents the main stages involved in constructing a fault tree, as well as specific adaptations relevant to information systems [3], [4].

The Fault Tree Analysis (FTA) method is a systematic analytical technique used to identify and evaluate the potential causes of undesired events or failures within complex systems. It allows for the logical modeling of cause-effect relationships and is widely applied in the fields of safety and reliability, particularly for high-impact critical systems. The primary objective of this method is to identify all possible combinations of faults or events that may lead to the main undesired outcome, referred to as the top event.

Thus, risks can be assessed, and preventive or corrective measures can be prioritized. The main operating principles are as follows:

- The method begins with a specified undesired event (for example, a system failure or a security breach).

- A reverse logical tree is then constructed, where the top event represents the root of the tree.
- The events and conditions that may lead to this top event are represented as nodes and are connected using logical operators (AND, OR, etc.).
- The basic events (fundamental elements that cannot be further decomposed into other causes) are identified.
- The analysis can be qualitative (determining the failure paths) or quantitative (calculating probabilities and frequencies).
- The construction of a fault tree involves a series of systematic steps designed to ensure a complete and accurate modeling of the causes leading to the top event:
- Definition of the top event: clearly establishing the malfunction or undesired incident to be analyzed.
- Identification of immediate causes: determining the events that can directly lead to the top event.
- Decomposition into intermediate and basic events: continuing the analysis until the fundamental events, which cannot be further broken down, are identified.
- Application of logical operators: connecting the events using logical operators (AND, OR, etc.) to describe the causal relationships.
- Validation of the fault tree: verifying the consistency and completeness of the model through expert consultation and critical review.

Within information systems, both fundamental and intermediate elements can be diverse and complex. A hardware malfunction [4], [5] may appear simple, yet when combined with a software error or human mistake, the resulting effects can be devastating. Each component-from servers to networks, from software code to operator procedures-must be regarded as a crucial element within this fragile network.

To provide an accurate and comprehensive representation, modeling does not stop at identifying causes but also incorporates their probabilistic assessment. This adds dimensions related to how vulnerable a system is to attacks, how frequently hardware or software errors may occur, and how effective the response to incidents proves to be.

This complex integration of factors enables not only an understanding of how and why failures occur, but also the anticipation and prevention of such events, thereby transforming the analysis into a vital tool for safeguarding and maintaining the functional integrity of highly critical systems.

3. Case Study - Fault Tree Analysis applied to a communication network within an organization

In communication networks, Fault Trees are employed to analyze the network's availability, reliability, and security. In such networks, cyberattacks can be modeled as basic events, representing various methods through which security may be compromised [5], [6], [7].

Examples:

- Unauthorized access: The attacker gains access to network resources through weak passwords, phishing, or other social engineering methods.
- DDoS attack (Distributed Denial of Service): Overloading network resources with false traffic to block legitimate user access.
- Software vulnerability exploitation: The attacker leverages known vulnerabilities (such as unpatched software) to execute malicious code.
- Data interception (Man-in-the-Middle): Capturing and manipulating network traffic.
- Malware and ransomware: Installing malicious programs that disrupt operations or compromise data.

Each type of attack can be considered an individual basic event in the Fault Tree.

Technical faults represent hardware or software issues that can affect the proper functioning of the network and may lead to service interruptions or performance degradation. Examples of such faults include:

- Hardware failure: Malfunction of routers, switches, servers, or other physical equipment.
- Software error: Bugs or faults in operating systems, firmware, or network applications.
- Connection loss: Physical disconnection or interruptions in cables, optical fibers, or wireless signals.
- Network overload: Excessive traffic leading to increased latency or packet loss.
- Configuration errors: Incorrect settings in devices or security policies.

These technical faults also represent basic events and can be combined with cyberattacks within the fault tree to evaluate the overall risk of the communication network.

Within an organization, the communication infrastructure is essential for the execution of daily operations. The network consists of a mix of hardware devices (routers, switches, servers) and software components (operating systems, applications, security policies), all interconnected and exposed to both technical and cybersecurity risks

To assess the overall risk of service disruption (considered the top event), a Fault Tree Analysis (FTA) approach was employed. Each type of attack or technical fault is treated as a basic event that may contribute to system failure.

The objective of this case study is to identify and model the primary potential causes that could lead to a complete IT service outage within the organization, with the goal of establishing priorities for preventive and corrective measures.

Accordingly, using Matlab software [8], a cyberattack is modeled as a Fault Tree. The probability of compromise is calculated both analytically (using formulas) and through Monte Carlo simulation, and the impact of changes in individual probabilities on the overall risk is analyzed. Additionally, the fault tree is visually represented. [9], [10],[11]

We consider the compromise event E0 =E1 OR (E2 AND E3), where E1 represents a successful phishing attack; E2 corresponds to the exploitation of a software vulnerability; represents unauthorized access (physical or remote) (Figure 1).

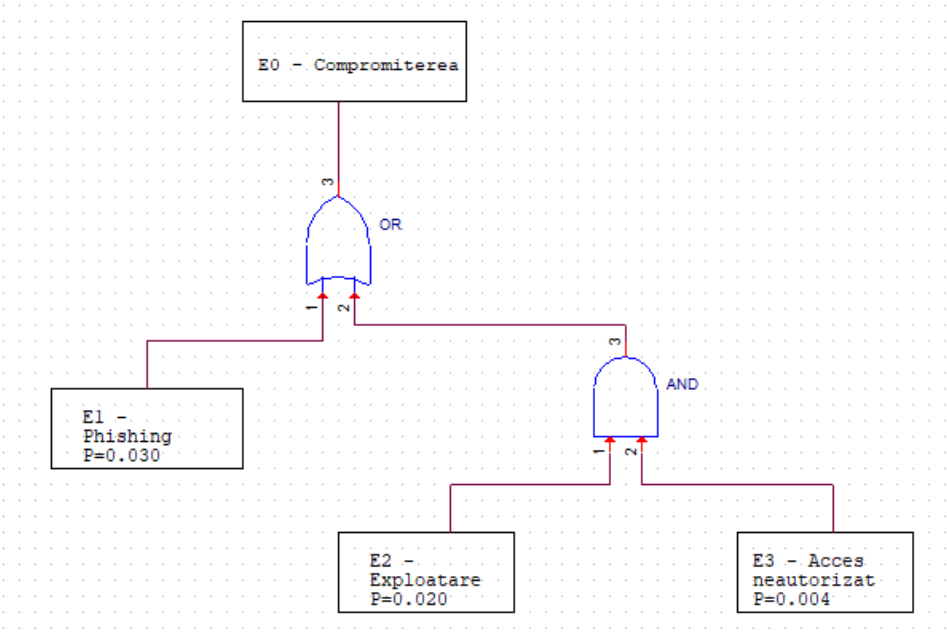

Fig. 1. Compromise Event

The basic probabilities for each elementary event are defined (Figure 2).

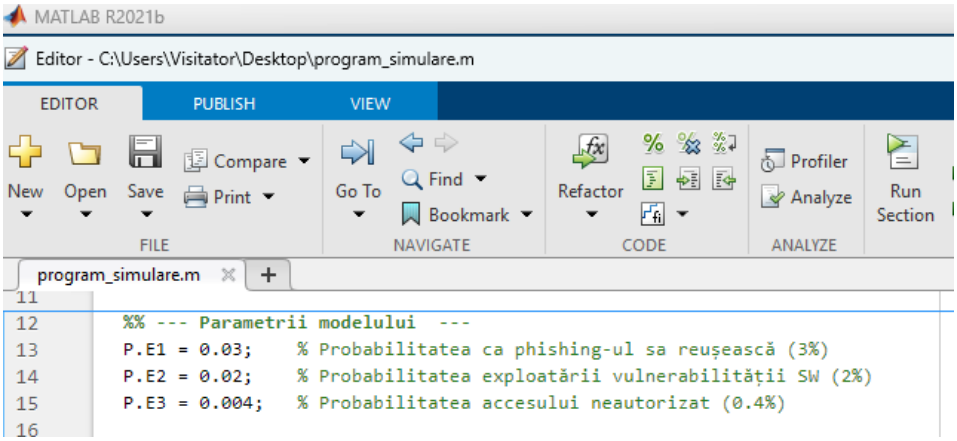

Fig. 2. Model Parameter Definition

The system is compromised if phishing occurs (E1) or if the software vulnerability is exploited while there is also unauthorized access (E2 AND E3).

Assuming the three events are independent, the combined probability for E0 (compromise) is calculated analytically as follows:

$$P(E2\ AND\ E3) = P(E2) \times P(E3) = 0.02 \times 0.004 = 0.00008$$

$$P(E0) = P(E1)\ OR\ P(E2\ AND\ E3)$$

$$P(E0) = 1 - \big(1 - P(E1)\big) \times \big(1 - P(E2) \times P(E3)\big) = 1 - (1 - 0.03) \times (1 - 0.00008)$$

$$\approx 0.3008$$

The probability of compromise is approximately 3.008%, which is nearly equal to the probability of phishing, since P(E2 AND E3) is much smaller (0.008%).

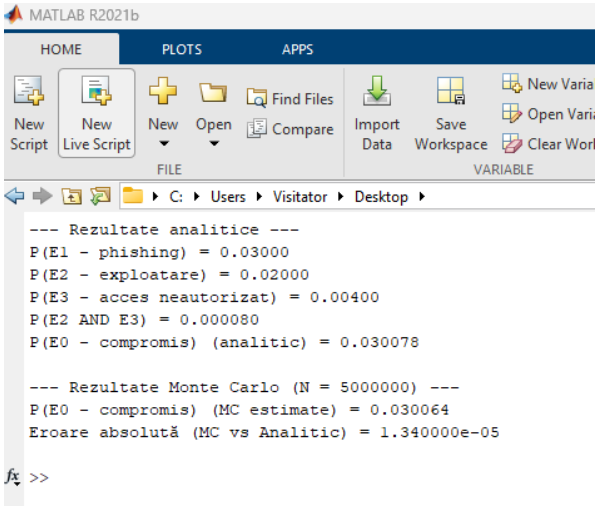

Fig. 3. Analytical Results and Monte Carlo Results

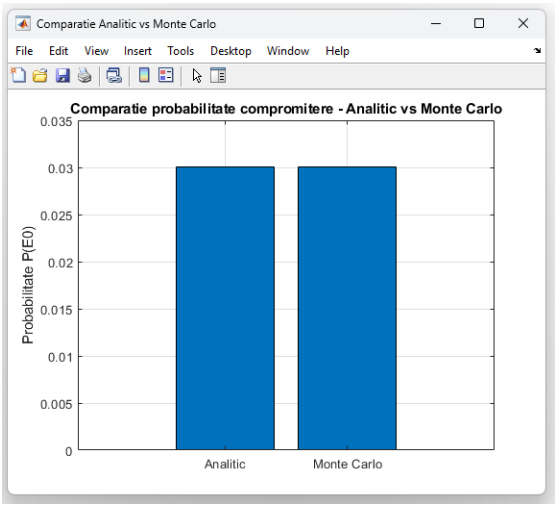

Fig. 4. Compromise Probability Comparisons - Analytical vs. Monte Carlo

Since the error between the Monte Carlo simulation and the analytical calculation is very small, it follows that the simulation correctly validates the model. The Monte Carlo estimate can be considered a verification of the analytical formula.

We further analyze how the probability of compromise, P(E0), varies when:

- P(E1) changes from 0 to 10%.
- P(E2) changes from 0 to 10%, keeping P(E3) constant.

If the curve for P(E1) is steeper, the system is more sensitive to phishing. If the curve for P(E2) is steeper, the software vulnerability has a greater impact. This allows us to identify which component of the fault tree is critical for the system's security (Figure 5).

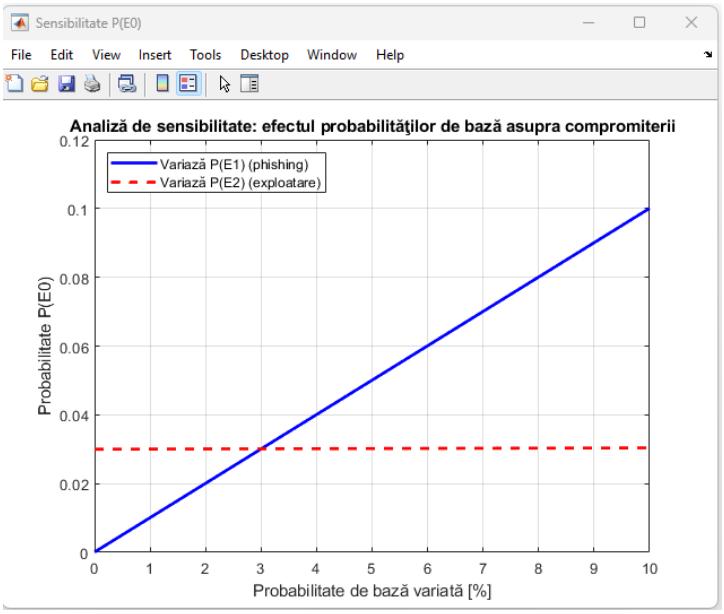

Fig. 5. Sensitivity Analysis

A simplified directed graph is constructed for the fault tree with the following nodes: E0 (compromise); OR (logic gate); E1 (phishing); AND (logic gate); E2 (exploitation); E3 (unauthorized access).

Figure 6 shows the fault tree graph, where the probabilities are also displayed next to the basic events.

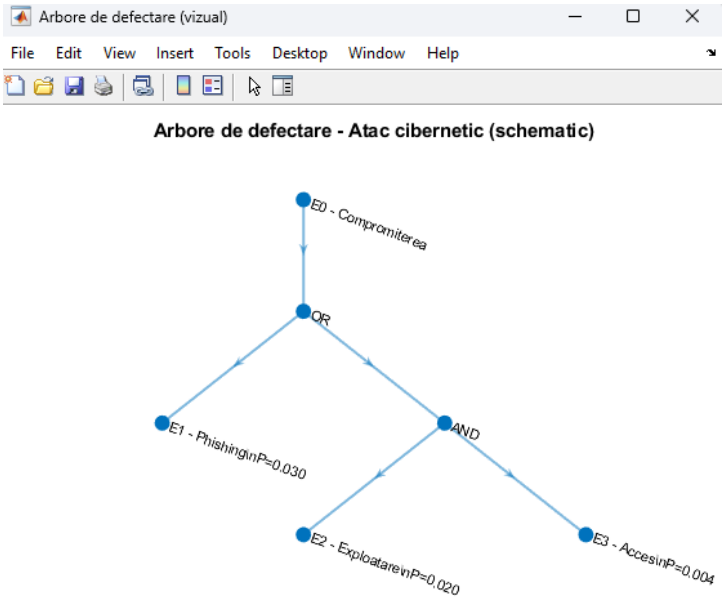

Fig. 6. Fault Tree - Cyberattack

From the conducted analyses, it is observed that the probability of compromise is dominated by phishing (E1), since P(E1) = 3% is much higher than the probability of E2 AND E3 = 0.008%.

Thus, the combined probability of exploiting the software vulnerability and unauthorized access is low, but still relevant enough to be considered in the overall system risk assessment.

The model provides an appropriate analytical framework for performing a sensitivity analysis, investigating the impact of variations in the individual probabilities of basic events (phishing and software exploitation) on the system's compromise probability.

The results obtained through Monte Carlo simulation offer a robust numerical validation of the analytical calculation, confirming the accuracy of the estimated probability for the top event based on the independence assumption.

The schematic graphical representation of the fault tree facilitates understanding of the logical relationships between basic events and how they combine to generate the top event, providing an intuitive view of the risk structure.

The following table presents, in a structured way, some protective measures that can be implemented according to each event [12], [13]:

Table 1. Protective Measures for Each Event

Event Code	Event Description	Probability	Protective Measures
E1	Phishing	0.03	- Anti-phishing training - Phishing simulations - Email filtering- MFA
E2	Exploitation of vulnerabilities	0.02	- Patch management - Vulnerability scanning - Intrusion prevention
E3	Unauthorized access	0.004	- Multi-Factor Authentication - Role-based access control - Logging & monitoring - Network segmentation
E0	System compromise	≈ 0.03008	- Integrated cybersecurity - Response plan - Periodic audits

4. Conclusions

This study has demonstrated the importance and effectiveness of the Fault Tree Analysis (FTA) method in assessing the security and reliability of high-functionality information systems (HFIS) used in critical sectors with significant impact on essential infrastructures and public safety.

By modeling cause-and-effect relationships logically, the fault tree allows for the clear identification of basic events and their combinations that may lead to the compromise of a critical system, providing a rigorous framework for risk assessment. The case study applied to a communication system highlighted how cyberattacks and technical failures can be integrated into a coherent probabilistic model.

Analytical results, obtained under the assumption of independent elementary events, were validated through Monte Carlo simulation, which confirmed the system compromise probability estimate with minimal error. This correlation strengthens confidence in the applicability of the method for quantitative security analysis.

Sensitivity analysis revealed that the risk of system compromise is dominated by the probability of phishing, while the combined probability of software vulnerability exploitation and unauthorized access, although less likely, remains an important factor that should not be overlooked in the overall risk assessment.

The schematic graphical representation of the fault tree facilitated understanding of the interdependencies among events and the logical structure of risks, serving as a valuable tool for communication and decision-making within security and reliability management processes.

Based on these findings, it is recommended to incorporate fault tree methodology into risk assessment and management strategies for critical information systems, as well as to extend the model by including dependencies between events and using advanced simulation techniques to improve estimation fidelity.

In conclusion, fault tree modeling proves to be a robust, adaptable, and relevant methodological tool for ensuring the security and reliability of high-functionality information systems, substantially contributing to the protection of essential infrastructures and the maintenance of operational continuity in critical contexts.

References

[1]. Modarres, M. (2016). Risk Analysis in Engineering: Techniques, Tools, and Trends. CRC Press.
[2]. Rausand, M. & Høyland, A. (2004). System Reliability Theory: Models, Statistical Methods, and Applications. Wiley-Interscience.
[3]. Kuo, W. & Zhu, D. (2013). Fault Diagnosis and Fault-Tolerant Control and Guidance for Aerospace Vehicles. Springer.
[4]. Kim, S. & Son, J. (2013). Fault Tree Analysis of Cybersecurity Attacks in Industrial Control Systems. International Journal of Security and Its Applications, 7(5), 65-76.
[5]. Bishop, M. (2003). Computer Security: Art and Science. Addison-Wesley.
[6]. ISO/IEC 27001:2013. Information Security Management Systems - Requirements. International Organization for Standardization.
[7]. Ross, R. & McEvilley, M. (2016). Systems Security Engineering: Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems. NIST Special Publication 800-160.
[8]. Higham, D. J. & Higham, N. J. (2016). MATLAB Guide (3rd ed.). SIAM.- Manual practic pentru programare și simulare în Matlab, incluzând metode Monte Carlo.
[9]. MathWorks Documentation. (n.d.). Fault Tree Analysis. Retrieved from https://www.mathworks.com/help/reliability/fault-tree-analysis.html.
[10]. Sham Tickoo. (2022). PCB Design Using OrCAD Capture and PCB Editor. CADCIM Technologies. ISBN: 978-1640571470.
[11]. Fishman, G. S. (1996). Monte Carlo: Concepts, Algorithms, and Applications. Springer.
[12]. Rubinstein, R. Y., & Kroese, D. P. (2016). Simulation and the Monte Carlo Method (3rd ed.). Wiley.
[13]. Pan, K., Liu, H., Gou, X., Huang, R., Ye, D., Wang, H., Glowacz, A. & Kong, J. (2022). Towards a Systematic Description of Fault Tree Analysis Studies Using Informetric Mapping. Sustainability, 14(18), 11430.