Proceedings of the 35th European Safety and Reliability & the 33rd Society for Risk Analysis Europe Conference
Edited by Eirik Bjorheim Abrahamsen, Terje Aven, Frederic Bouder, Roger Flage, Marja Ylönen
©2025 ESREL SRA-E 2025 Organizers. *Published by* Research Publishing, Singapore.
doi: 10.3850/978-981-94-3281-3_ESREL-SRA-E2025-P1598-cd

Safety and Security Risk Mitigation in Satellite Missions via Attack-Fault-Defense Trees

Reza Soltani

Formal Methods and Tools group, University of Twente, Enschede, the Netherlands.
E-mail: r.soltani@utwente.nl

Pablo Diale

Ascentio Technologies S.A., Cordoba, Argentina. E-mail: pdiale@ascentio.com.ar

Milan Lopuhaä-Zwakenberg

Formal Methods and Tools group, University of Twente, Enschede, the Netherlands.
E-mail: m.a.lopuhaa@utwente.nl

Mariëlle Stoelinga

Formal Methods and Tools group, University of Twente, Enschede, the Netherlands;
Radboud University, Nijmegen, the Netherlands. E-mail: m.i.a.stoelinga@utwente.nl

Cyber-physical systems, such as self-driving cars or digitized electrical grids, often involve complex interactions between security, safety, and defense. Proper risk management strategies must account for these three critical domains and their interaction because the failure to address one domain can exacerbate risks in the others, leading to cascading effects that compromise the overall system resilience. This work presents a case study from Ascentio Technologies, a mission-critical system company in Argentina specializing in aerospace, where the interplay between safety, security, and defenses is critical for ensuring the resilience and reliability of their systems. The main focus will be on the Ground Segment for the satellite project currently developed by the company. Analyzing safety, security, and defense mechanisms together in the Ground Segment of a satellite project is crucial because these domains are deeply interconnected—for instance, a security breach could disable critical safety functions, or a safety failure could create opportunities for attackers to exploit vulnerabilities, amplifying the risks to the entire system. This paper showcases the application of the Attack-Fault-Defense Tree (AFDT) framework, which integrates attack trees, fault trees, and defense mechanisms into a unified model. AFDT provides an intuitive visual language that facilitates interdisciplinary collaboration, enabling experts from various fields to better assess system vulnerabilities and defenses. By applying AFDT to the Ground Segment of the satellite project, we demonstrate how qualitative analyses can be performed to identify weaknesses and enhance the overall system's security and safety. This case highlights the importance of jointly analyzing attacks, faults, and defenses to improve resilience in complex cyber-physical environments.

Keywords: cyber-physical systems, Fault Trees, Attack Trees, Defense, Risk Mitigation, Safety, Security.

1. Introduction

In recent years, the satellite industry has witnessed a paradigm shift with the emergence of Ground Segment as a Service (GSaaS), which enables satellite operators to outsource ground segment operations, including data processing, mission control, and communication infrastructure, to specialized service providers. This innovative model enables satellite operators to access and manage ground segment infrastructure on a pay-as-you-go basis, eliminating the need for substantial capital investments in proprietary ground systems. By leveraging GSaaS, operators can expedite mission deployment, reduce operational costs, and concentrate on core business activities such as data provision and analysis.

However, the adoption of GSaaS introduces new challenges, particularly in ensuring the safety and security of satellite operations. The shared nature of GSaaS infrastructure necessitates robust mechanisms to protect against potential threats and system failures.

In cyber-physical systems, safety and security are frequently investigated separately in different studies. Nevertheless, there is a strong interdependency between them (Nicoletti et al., 2023). In the complex safety and security interplay that involves trade-offs, measures that improve security may weaken safety or vice versa. We need to increase the resilience of critical infrastructures such as GSaaS for satellite operators not only to accidental failures that may come from many high-tech components but also to (cyber)attacks by malicious actors. To achieve high resilience against such risks, we may consider using countermeasures against safety and security risks. However, experts from several fields must collaborate on such implementation, which leads to the need to have a common framework for assessing the safety, security, and impact of countermeasures.

Tree-based models are ubiquitous in both safety and security risk assessment. Fault Trees (FTs) (IEC, 2006; Lee et al., 1985) are introduced for safety, and Attack Trees (ATs) (Schneier, 1999) for security. These are often used frameworks for allowing communication across disciplines. To capture the wide range of risks and associated countermeasure strategies, more comprehensive models are required, as FTs and ATs only address safety and security, respectively. There are frameworks for joint analysis like Attack-Defense Trees (ADTs), which model security risks and countermeasures to mitigate them, and Attack-Fault Trees (AFTs), which represent joint safety-security risks. However, none of these models has the expressive power to model the interaction between safety, security, and defense that is inherent to critical infrastructures such as GSaaS, particularly in scenarios where system failures, cyberattacks, and defensive measures interact in complex and unpredictable ways.

To address these concerns, we introduced a new framework, namely Attack-Fault-Defense Trees (AFDTs), that captures all safety, security, and defense domains in a single framework (Soltani et al., 2024). In our previous work (Soltani et al., 2024), we presented the mathematical definition of AFDTs and their structure-function along with the semantics and cut set metrics. In addition, we provided a case study of a power grid to showcase the application of our framework (Soltani et al., 2024). While that study primarily emphasized safety aspects, aligning with the domain and requirements of that specific application, it also had a limited number of defenses. We address these limitations by presenting a new case study and applying AFDTs to a GSaaS environment, which is inherently more security-dominant, that works better for AFDTs with many defenses. We also perform qualitative and quantitative risk analyses, showcasing the scalability and applicability of AFDTs in enhancing the resilience of satellite ground segment services.

2. Related Work

In the safety and security domain, tree-based formalisms form the majority of formalisms that capture the interplay between security and safety (Nicoletti et al., 2023). Attack Trees (ATs) (Schneier, 1999) deal with system attacks, while Fault Trees (FTs) (IEC, 2006; Lee et al., 1985) were made to handle system failures. In a survey of models for safety-security co-analysis, Nicoletti et al. (2023) discovered that there is no model that precisely represents safety-security interactions. Instead, a variety of methods are used to combine constructs from frameworks that only concentrate on security or safety. Metrics are not distinct; not one is uniquely designed with safety/security interactions in mind. Furthermore, there is a shortage of large-scale case studies, and current formalisms only model dependencies in small- and medium-sized case studies.

Due to differences in how they are used, FTs and ATs are extended either with additional gates and system recovery (Čepin and Mavko, 2002; Roy et al., 2012) or defenses (Kordy et al., 2014,?). Kordy et al. (2014) define attack–defense trees as attack trees with defenses in the form of countermeasures. Fila and Wideł (2020) look into the most effective countermeasures for ADTs. The techniques used by Khouzani et al. (2019) to determine the optimal countermeasures in attack graphs, an alternative risk model for security, rely

Table 1.: Related works comparison to the proposed approach

Model		Attack	Failure	Defense/ Countermeasure	Qualitative/Quantitative analysis	Case study
FT	Čepin and Mavko (2002)		✓		✓	
ADT	Kordy et al. (2010)	✓		✓		
	Kordy et al. (2014)	✓		✓	✓	
	Roy et al. (2012)	✓		✓	✓	✓
	Fila and Wideł (2020)	✓		✓	✓	✓
AT	Khouzani et al. (2019)	✓		✓	✓	✓
AFDT	Soltani et al. (2024)	✓	✓	✓	✓	✓

on the activities of each defender, which affects the probability that an attack will be successful. Sabaliauskaite and Mathur (2015) developed a failure-attack-countermeasure graph architecture to align safety and security during the early stages of the development of cyber-physical systems. Safety, security, and countermeasures are only included in the graph in the early stages of development. The paper does not include a semantic or qualitative analysis of the graph.

We compared related work to the AFDT approach in Table 1. As can be seen, AFDT is unique in combining failures, attacks, and defenses. In our previous work (Soltani et al., 2024), we showcased the application of the AFDT framework through a safety-dominant case study, which primarily focused on safety with a limited number of defenses. In this paper, we extend our approach to a more security-dominant case study, applying AFDT to a GSaaS environment for satellite operations, which incorporates a greater number of defenses to address its distinct challenges.

3. Background: AFDT

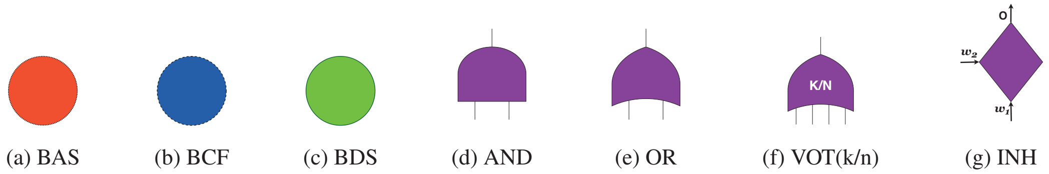
AFDT. The various elements of AFDT (Soltani et al., 2024) are illustrated in Figure 1, with Figs. 1a–1c depicting the various leaf types as circular nodes, such as the Basic Attack Step (BAS), Basic Component Failure (BCF), and Basic Defense Step (BDS). The interdependence between leaves happens through logical symbols called gates. These gates mirror (illustrate) discrete events and have inputs from other gates or individual leaves. We considered the gate types *AND* (Fig. 1d), *OR* (Fig. 1e), *VOT(k/n)* (voting, Fig. 1f), and *INH* (inhibition, Fig. 1g) in the analysis, that activate when resp. all, one, and k of the N inputs are activated. The integrity of the system is compromised when the TLE is activated.

When both BCFs and BASs are present, the model is an Attack-Fault Tree (AFT). AFT models both safety and security and their interaction. An example of an AFT is shown in Fig. 2 with two BCFs and two BASs. The TLE is triggered when both the component failure C_1 and the attack A_1 occur, or when the failure C_2 coincides with the attacks A_1 and A_2, under the condition that at least two of these three events must occur.

AFDTs, introduced in our previous work (Soltani et al., 2024), extend AFTs by incorporating defenses to prevent the propagation of safety and security risks. This formalism unifies the strengths of AFT and ADT, allowing for a comprehensive analysis of faults, attacks, and defenses within a single framework. See Fig. 3 for a visual comparison of tree-based formalisms.

Compared to AFT, AFDT introduces two key elements: the Basic Defense Step (BDS), representing atomic actions by the defender to enhance system resilience, and the *INH*-gate, which models countermeasures that prevent specific events from propagating when triggered. Figure 4 illustrates how defenses (e.g., D_1 and D_2) are added to the AFT in Figure 2, showcasing their role in mitigating risks such as C_1 with A_1, and the joint propagation of A_1, A_2, and C_2. Disabling relations, modeled by *INH*-gates, capture scenarios where defenses can fail due to components like C_3. Full formal details of AFDTs are available in Soltani et al. (2024).

Qualitative analysis. AFDTs provide a complete, system-wide view of how safety and security hazards might merge to cause system-level failures. It depicts how attacks and failures propa-



(a) BAS (b) BCF (c) BDS (d) AND (e) OR (f) VOT(k/n) (g) INH

Fig. 1.: Attack-Fault-Defense tree elements

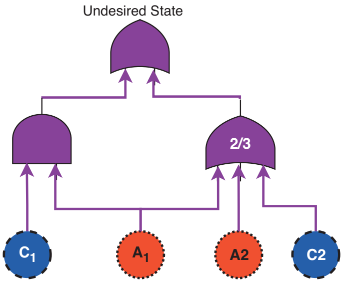


Fig. 2.: An AFT example. The system fails if either C_1 and A_1 both occur, or if at least 2 of A_1, A_2, C_2 occur.

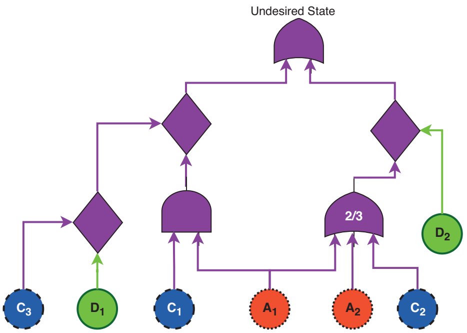


Fig. 4.: An exemplary depiction of AFDT, which extends Fig. 2 by defenses D_1 and D_2

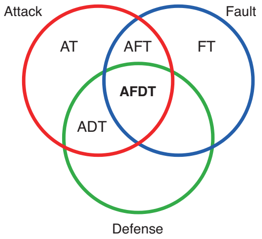


Fig. 3.: Venn diagram representing the tree-based formalisms discussed in this paper.

Table 2.: MCS analysis of the toy example AFDT

No defense	$\{D_1\}$	$\{D_2\}$	$\{D_1, D_2\}$
$\{C_1, A_1\}$	$\{C_3, C_1, A_1\}$	$\{C_1, A_1\}$	$\{C_3, C_1, A_1\}$
$\{A_1, A_2\}$	$\{A_1, A_2\}$	✗	✗
$\{A_1, C_2\}$	$\{A_1, C_2\}$	✗	✗
$\{A_2, C_2\}$	$\{A_2, C_2\}$	✗	✗

gate to a higher level, resulting in the failure of a top-level event (TLE), and how defenses prevent this propagation. Finding the Minimal Cut Sets (MCSs) for AFTs is one of the most used methods of qualitative risk analysis. An MCS is a minimum collection of BCFs/BASs that, when added together, activate the TLE; if any of these elements are removed, the TLE becomes inactive. Consider the AFT in Fig. 2. It contains four MCSs: $\{C_1, A_1\}$, $\{A_1, A_2\}$, $\{A_1, C_2\}$, and $\{A_2, C_2\}$. Any of these MCSs gives information about the system's vulnerability and also depicts the most concise path that caused the activation (failure) of the TLE. Each MCS has a finite number of elements.

An AFDT's identification of MCSs is different from an AFT's because the former has protections that can prevent an attack from propagating. We have four sets of MCSs in the AFDT of Fig. 4 since there are two BDSs. An overview of the MCSs corresponding to each defense activation for AFDT of Fig. 4 is provided in Table 2. Table 2 indicates that the defense D_2 activation results in the removal of three MCSs, specifying its effectiveness. Defense D_1 increases the element count of one MCS but has no effect on another. System dependability is increased when both defenses are activated at the same time, creating a three-element MCS. It should be noted that increasing the size of an MCS, as well as deleting it entirely, increases the reliability of a system. MCS analysis in AFDT provides an overview of each defense's influence and relationship to system reliability. We can analyze MCS quantitively when data is available by considering the defense's effect on MCS parameters, such as probability.

4. Satellite Ground Segment use-case

In this case study, we focus exclusively on the ground segment of satellite operations within the context of the Ground Segment as a Service (GSaaS) paradigm. The ground segment (GS) plays a critical role in satellite operations, acting as the interface between the space segment and the end users. The GS handles crucial operations, including telemetry, tracking, command, and the acquisition of mission data. Furthermore, in a GSaaS model, the GS's cloud-based nature introduces unique opportunities and challenges that warrant targeted analysis. Unlike traditional GSs, which are tightly integrated with proprietary satellite systems, GSaaS decouples the ground infrastructure and offers scalable, on-demand services. This abstraction allows operators to focus on mission objectives while leveraging the flexibility and cost efficiency of shared infrastructure.

The GSaaS under consideration in this case study is a cloud-based solution proposed by Ascentio Technologies S.A., a company with extensive experience in satellite ground infrastructure. Ascentio's GSaaS platform aims to provide satellite operators with seamless access to ground stations (while also allowing the users to interface with their own stations), enabling them to execute uplink and downlink operations, process mission data, and monitor satellite health via cloud-native interfaces. This architecture significantly reduces the overhead associated with building and maintaining dedicated ground infrastructure while offering the scalability required to support multi-mission operations.

The case study evaluates the safety and security of Ascentio's GSaaS approach using the proposed AFDT methodology. The analysis begins by identifying potential vulnerabilities inherent to cloud-based GSaaS platforms, including risks related to data breaches and service disruptions. By mapping these vulnerabilities into the AFDT framework, we aim to systematically assess the interactions between security threats, system faults, and implemented defense mechanisms. The resulting AFDT diagram serves as a comprehensive visual representation of risks and mitigations, providing actionable insights for strengthening the platform's resilience.

Another critical aspect of the analysis is the identification of MCSs. These cut sets offer valuable information for prioritizing risk mitigation strategies and optimizing resource allocation. A detailed discussion of the analysis of MCSs will be included in a subsequent subsection.

By focusing on the ground segment of Ascentio's cloud-based GSaaS design, this case study provides a clear and detailed exploration of the paradigm's implications for safety and security. The AFDT methodology is applied to uncover latent risks, assess defense effectiveness, and propose strategies to enhance the robustness of satellite ground infrastructure in this novel service model. This analysis not only contributes to the ongoing development of Ascentio's platform but also provides a blueprint for applying AFDTs to other GSaaS implementations as well as other traditional GS infrastructures. Fig. 5 depicts the AFDT for the Ascentio's cloud-based GSaaS.

4.1. *Application of AFDT to a GSaaS*

The AFDT developed for this case study reflects the unique risks and mitigation strategies associated with the cloud-based GS infrastructure, emphasizing its critical functions: telecommand transmission and telemetry reception.

Structure and Focus. The AFDT centers on the top-level event (TLE) defined as the failure to ensure the correct and reliable execution of telecommand and telemetry operations. This TLE branches into intermediate events, key failure pathways specific to the GSaaS environment:

- Ground Station Unavailability:
  Events leading to a loss of communica-

tion between ground segment and satellite, such as hardware malfunctions, connectivity issues, or signal interference.

- Faulty Command or Telemetry Data: Situations where incorrect, incomplete, or corrupted telecommands or telemetry data compromise system integrity.
- Human or Procedural Errors: Mistakes in scheduling, command generation, or manual operations that propagate into larger failures.

Iterative Development and Refinement. Initial iterations focused on cataloging typical failure modes in GSaaS environments, which were then enriched by analyzing real-world cases of cloud-based system failures. Intermediate events and their dependencies were refined to model how faults and attacks propagate through the GSaaS architecture. For instance, a DDoS attack on the API endpoints of a virtualized application could lead to service unavailability. This vulnerability is mitigated in the AFDT by modeling defenses such as cloud-based DDoS protection services.

4.2. *Quantitative analysis of GSaaS*

As specified in Section 3, MCSs provide information regarding the system's vulnerability and define the most concise path that causes TLE failure. This study presents a qualitative approach to analyzing the system through the identification of MCSs and their most effective defenses. Examining MCSs gives valuable insights into the system's reliability, as the system's overall reliability can be augmented by mitigating the attack or failure probability associated with these MCSs. By mapping MCSs to their corresponding defenses, the proposed method provides a clear and systematic overview of potential vulnerabilities and mitigation strategies. Even in the absence of quantitative data, this approach proves to be a valuable tool for gaining insights into the interplay between system failures, attacks, and defenses, facilitating better-informed decision-making for enhancing system safety and security.

Table 3 indicates the MCSs related to the GSaaS AFDT of Fig. 5. The table highlights the effectiveness of various defenses in mitigating the MCSs identified for the GSaaS AFDT. Each defense mechanism eliminates the associated MCSs, enhancing the system's overall resilience. For instance, the *E2E* defense is effective against the *MITM* attack, while *DP* mitigates the risk posed by *DDoS*. Similarly, *Seg* is a crucial defense strategy that addresses a wide range of MCSs involving multiple *AS* nodes.

Especially interesting is the defense *TSA*, which protects against both the BCF *Bug* and the BAS *SCA*; and the MCS $\{Pass, Uname, HE\}$, which consists of both BCFs and BASs. This highlights the importance of analyzing safety, security, and countermeasures in unison.

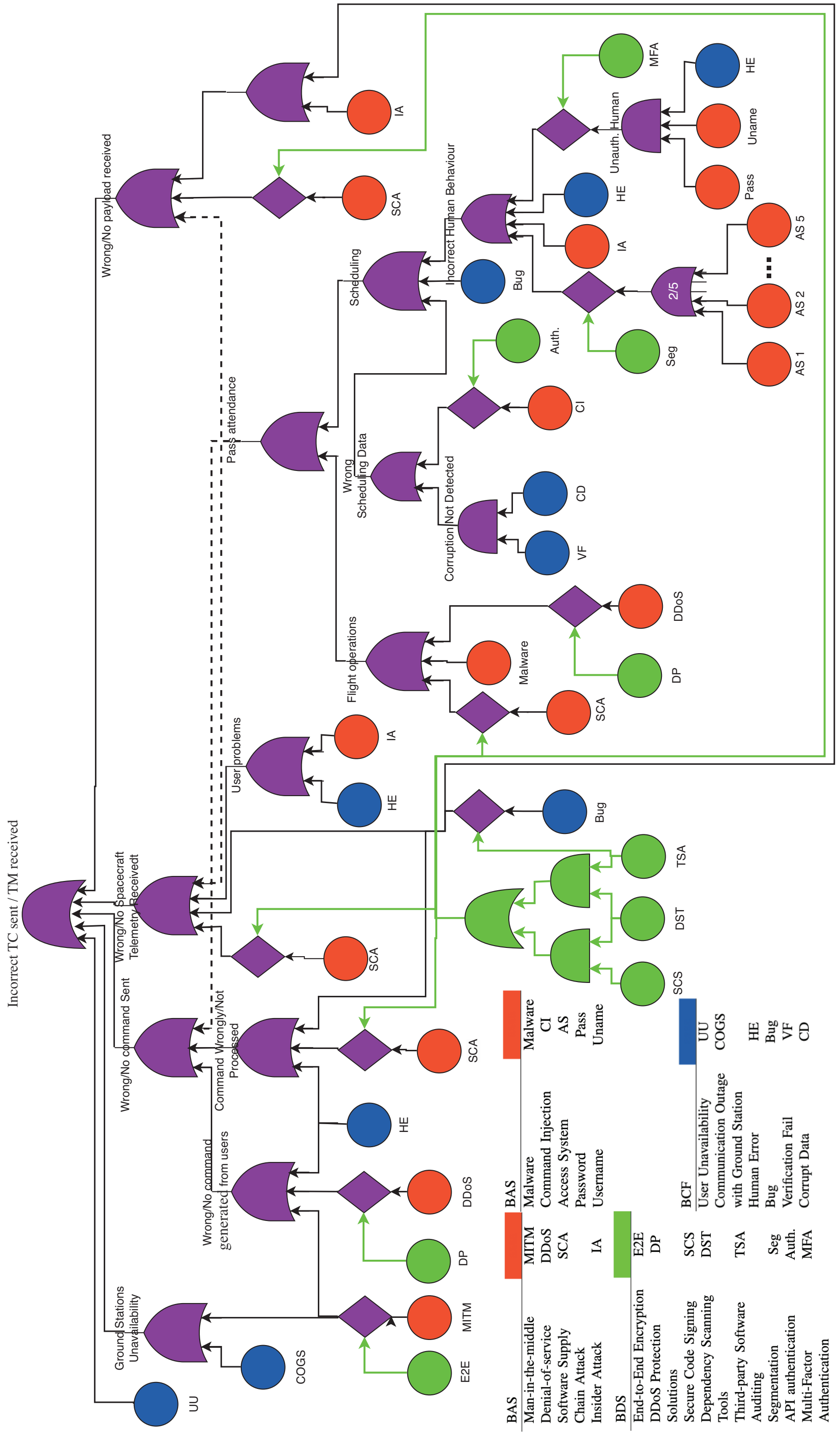
Certain failures or attacks remain defenseless. This highlights potential vulnerabilities in the system that require further attention to ensure comprehensive risk mitigation. Overall, the analysis of this table underscores the need for a balanced and comprehensive approach to implementing defenses that can effectively mitigate both safety and security risks while identifying and addressing gaps in the current defense mechanisms.

5. Conclusion

This case study underscores the critical value of using the AFDT framework to address the intertwined challenges of safety, security, and defense in complex cyber-physical systems. By applying AFDT to the ground segment of a satellite project, we demonstrated how a unified approach identifies vulnerabilities, assesses risks, and guides the development of robust defense strategies, ultimately enhancing the system's overall resilience and reliability. The presented model aids experts from different fields in uncovering complex dependencies w.r.t. safety and security and analyzing how innovations may affect the security and safety of the intertwined system using MCS analysis.

Acknowledgement

The work reported in this paper was partially funded by the NWO under grants PrimaVera (NWA. 1160. 18. 238), ZORRO (KICH1. ST02. 21. 003), Marie Skłodowska-Curie grant MISSION (101008233), ERC under grants CAESAR (864075), and RUBICON (101187945).



BAS	
Man-in-the-middle	MITM
Denial-of-service	DDoS
Software Supply Chain Attack	SCA
Insider Attack	IA

BDS	
End-to-End Encryption	E2E
DDoS Protection Solutions	DP
Secure Code Signing	SCS
Dependency Scanning Tools	DST
Third-party Software Auditing	TSA
Segmentation	Seg
API authentication	Auth.
Multi-Factor Authentication	MFA

BAS	
Malware	Malware
Command Injection	CI
Access System	AS
Password	Pass
Username	Uname

BCF	
User Unavailability	UU
Communication Outage with Ground Station	COGS
Human Error	HE
Bug	Bug
Verification Fail	VF
Corrupt Data	CD

Fig. 5.: The AFDT of GSaaS

Table 3.: MCS analysis of the GSaaS AFDT

MCS	Effective defense(s)
{UU}	∅
{COGS}	∅
{MITM}	{E2E}
{DDoS}	{DP}
{HE}	∅
{SCA}	{SCS, DST}, {DST, TSA}
{IA}	∅
{Bug}	TSA
{Malware}	∅
{VF, CD}	∅
{CI}	{Auth.}
{AS1, AS2}	{Seg}
{AS1, AS3}	{Seg}
{AS1, AS4}	{Seg}
{AS1, AS5}	{Seg}
{AS2, AS3}	{Seg}
{AS2, AS4}	{Seg}
{AS2, AS5}	{Seg}
{AS3, AS4}	{Seg}
{AS3, AS5}	{Seg}
{AS4, AS5}	{Seg}
{Pass, Uname, HE}	{MFA}

References

Fila, B. and W. Wideł (2020). Exploiting attack–defense trees to find an optimal set of countermeasures. In *2020 IEEE 33rd Computer Security Foundations Symposium (CSF)*, pp. 395–410.

IEC, I. (2006). 61025: Fault tree analysis (fta). Technical report, Technical Report.

Khouzani, M., Z. Liu, and P. Malacaria (2019). Scalable min-max multi-objective cyber-security optimisation over probabilistic attack graphs. *European Journal of Operational Research 278*(3), 894–903.

Kordy, B., S. Mauw, M. Melissen, and P. Schweitzer (2010). Attack–defense trees and two-player binary zero-sum extensive form games are equivalent. In T. Alpcan, L. Buttyán, and J. S. Baras (Eds.), *Decision and Game Theory for Security*, Berlin, Heidelberg, pp. 245–256. Springer Berlin Heidelberg.

Kordy, B., S. Mauw, S. Radomirović, and P. Schweitzer (2014). Attack–defense trees. *Journal of Logic and Computation 24*(1), 55–87.

Kordy, B., L. Piètre-Cambacédès, and P. Schweitzer (2014). Dag-based attack and defense modeling: Don't miss the forest for the attack trees. *Computer science review 13*, 1–38.

Lee, W.-S., D. L. Grosh, F. A. Tillman, and C. H. Lie (1985). Fault tree analysis, methods, and applications a review. *IEEE transactions on reliability 34*(3), 194–203.

Nicoletti, S. M., M. Peppelman, C. Kolb, and M. Stoelinga (2023). Model-based joint analysis of safety and security: Survey and identification of gaps. *Computer Science Review 50*, 100597.

Roy, A., D. S. Kim, and K. S. Trivedi (2012). Attack countermeasure trees (ACT): towards unifying the constructs of attack and defense trees. *Security and Communication Networks 5*(8), 929–943.

Sabaliauskaite, G. and A. P. Mathur (2015). Aligning cyber-physical system safety and security. In M.-A. Cardin, D. Krob, P. C. Lui, Y. H. Tan, and K. Wood (Eds.), *Complex Systems Design & Management Asia*, Cham, pp. 41–53. Springer International Publishing.

Schneier, B. (1999). Modeling security threats. *Dr. Dobb's journal 24*(12).

Soltani, R., M. Lopuhaä-Zwakenberg, and M. Stoelinga (2024). Safety-security analysis via attack-fault-defense trees: Semantics and cut set metrics. In A. Ceccarelli, M. Trapp, A. Bondavalli, and F. Bitsch (Eds.), *Computer Safety, Reliability, and Security*, Cham, pp. 218–232. Springer Nature Switzerland.

Soltani, R., B. Ozceylan, M. Lopuhaä-Zwakenberg, C. Kolb, and G. Hoogsteen (2024). Safety and security dependencies for gridshield.

Čepin, M. and B. Mavko (2002). A dynamic fault tree. *Reliability Engineering & System Safety 75*(1), 83–91.