International Journal of Information Security (2025) 24:169
https://doi.org/10.1007/s10207-025-01082-4

REGULAR CONTRIBUTION

xT-STRIDE threat model for unmanned air vehicle security

Aytac Ugur Yerden[1] · **Sinan Senol**[2] · **Mehmet Kara**[3] · **Savas Dilibal**[1]

Accepted: 8 June 2025

Abstract

In recent years, substantial progress has been made in the design and development of unmanned aerial vehicles (UAVs) with diverse functional capabilities for avionic applications. Beyond addressing performance bottlenecks, the growing threat of cybersecurity attacks introduces additional complexity to the UAV design process. Developing a high-performance UAV system involves navigating technical security challenges that must be addressed pragmatically. Implementing excessive security measures can increase costs and degrade system performance, while insufficient protection of vulnerable components may expose the UAV to potential exploitation. This study introduces a multi-layered drone system, thoroughly examines threats to its components, and outlines corresponding safeguards in trusted layers. The security requirements for UAV systems were defined using the Common Criteria framework. To strengthen UAV security under real-world operational conditions, a novel threat modeling approach, xT-STRIDE, has been developed. This method incorporates the UAV's communication and embedded system architecture, modeled with realistic security elements. The xT-STRIDE framework was evaluated theoretically in conjunction with traditional threat modeling techniques.

Keywords UAV · Internet of Things · Cybersecurity · Threat modeling · STRIDE

1 Introduction

Unmanned aerial vehicles (UAVs), commonly known as drones, have become increasingly prominent in modern life owing to their wide range of applications. They offer numerous benefits across civilian sectors, contribute to public safety, enhance military capabilities, and can even be used for offensive purposes. In the context of smart cities and sustainable development, various types of UAVs play vital roles in collecting, analyzing, and delivering highly accurate and sometimes classified data [1]. A comprehensive review underscores their diverse applications in areas such as traffic control, construction, photogrammetry, remote sensing, and transportation. It also highlights the importance of effective navigation, energy efficiency, and the extraction of critical information to ensure the safe and optimized use of UAVs in transportation systems [2]. The critical functionality of UAVs highlights the potentially catastrophic consequences of cyberattacks targeting these systems. As next-generation UAVs take on increasingly diverse tasks, the risk of malicious attacks remains a considerable concern. Equipped with data-collecting sensors, UAVs are susceptible to external threats that may access sensitive sensory data or inject false information. Given their autonomous control capabilities, such attacks can result in unexpected and potentially irreversible damage [3]. This highlights the critical need for thorough safety and security assessments to support the widespread adoption of UAVs. UAV safety considerations include risks to airspace, individuals, and ground infrastructure. Key issues to address involve UAV collisions, limitations in current air traffic management systems, equipment and sensor failures, adverse weather conditions, and potential harm to buildings and people at low altitudes. The security aspect focuses on protecting various types of data, including images, biometric

✉ Aytac Ugur Yerden
aytac.yerden@gedik.edu.tr

Sinan Senol
ssenol5@ford.com.tr

Mehmet Kara
mehmet.kara@kocaelisaglik.edu.tr

Savas Dilibal
savas.dilibal@gedik.edu.tr

1 Faculty of Engineering, Mechatronics Engineering Department, Istanbul Gedik University, Istanbul, Turkey

2 Ford Otosan Inc, 34885 Kartal, Istanbul, Turkey

3 Faculty of Engineering and Natural Sciences, Computer Engineering Department, Kocaeli Health and Technology University, Kocaeli, Turkey

Published online: 01 July 2025

information, audio recordings, and location details. A major concern is ensuring that the data captured by UAVs remain confidential and are not compromised by unauthorized external entities [4]. The wide range of connectivity features in UAVs exposes multiple potential attack vectors that adversaries could exploit.

Implementing robust security measures is essential to mitigate potential risks associated with the misuse of UAVs, given their wide range of applications. These risks include unauthorized disclosure of sensitive information, tampering, and other harmful consequences. Ensuring the secure design of UAVs is currently a critical focus for both academic researchers and standardization bodies. However, developing security standards for UAV design presents considerable challenges owing to the broad diversity of hardware and software components involved. Furthermore, the operational environments in which UAVs are deployed vary widely, adding complexity to standardization efforts. Owing to the unique architecture and operational constraints of UAVs, traditional information technology (IT) security standards, frameworks, and models—such as ISO 27001, Microsoft's Secure Software Development Life Cycle (SDLC), the Open Web Application Security Project (OWASP), the Software Assurance Maturity Model (SAMM), and the Capability Maturity Model Integration (CMMI)—cannot be directly applied to model trust levels in UAV systems. The Common Criteria standard provides a foundational framework for defining and specifying the necessary security requirements for UAVs by offering multiple assurance levels. UAVs are deployed across a wide range of domains, including exploration, surveillance, package delivery, assistance, and emergency response—each carrying critical security and operational implications. To ensure security and privacy, it is essential to safeguard confidentiality, integrity, and availability—particularly when processing, storing, and transmitting sensitive data. In smart cities, critical activities such as monitoring traffic patterns, fire incidents, environmental conditions, and disaster response involve the handling of confidential information. Therefore, implementing effective precautionary measures is essential to reduce the risk of data misuse.

The xT-STRIDE methodology for threat analysis and risk assessment in UAV cybersecurity is designed to protect assets in UAV systems. Building upon existing frameworks, this approach highlights the critical role of component selection in ensuring system security [5–9]. Users can choose between secure-by-design components—referred to as trusted components—or components lacking inherent security, which require additional engineering to address identified vulnerabilities. In practice, secure-by-design engineering involves assessing assets, their vulnerabilities, their value, and the external interactions that could introduce threats. Trusted components help minimize the external threat surface, while non-secure components considerably expand the attack surface and demand targeted security measures. xT-STRIDE is designed to provide a realistic assessment of the attack surface in UAV systems, with a focus on identifying vulnerable components. Building on the STRIDE threat modeling framework, this methodology systematically identifies and addresses potential vulnerabilities. Additionally, xT-STRIDE is flexible and can be integrated with other threat modeling approaches, allowing for the customization of trusted asset categorization. While this study defines the asset space in the application development stack, it can also be adapted to include electronic components or system functions. By prioritizing the identification and mitigation of high-probability vulnerabilities, xT-STRIDE aims to strengthen cybersecurity efforts in UAV systems.

xT-STRIDE is based on the STRIDE framework and introduces an additional trust layer [10]. This trust layer is essential for the design of system components because it allows for a thorough assessment of the various subcomponents in each device's engineering. Given the complexity of this process, a systematic approach is required to design these sub-components effectively. The trust level technique addresses this challenge by leveraging the Common Criteria [11] evaluation to assess and determine the appropriate security level.

Modern UAV systems rely heavily on their communication networks to enable effective cooperation and coordination among multiple units. In operational environments where diverse cyber threats are prevalent, maintaining cooperative capabilities among UAVs becomes critically important [12]. Previous studies primarily focused on securing collaboration between UAVs through layered security measures, particularly emphasizing physical controls [13]. In contrast, our current study takes a more holistic approach by addressing the full spectrum of cybersecurity. The smooth operation of UAVs depends not only on their ability to collaborate but also on the essential reliability of their network connections. To address this issue, a previous study [14] proposes a strategy specifically designed for UAVs, emphasizing the reliability of their connections. The study explores the security implications of integrating drones into an Internet of Vehicles (IoV) system, using trust- and priority-based algorithms. The proposed approach incorporates modules for both drone-to-drone (D2D) and drone-to-vehicle (D2V) communication to optimize data routing efficiency. While previous studies have addressed similar challenges, our methodology provides a more comprehensive and inclusive solution. It is important to recognize that these findings highlight the growing emphasis in the community on building robust and efficient networks. However, a thorough threat analysis is essential, as highlighted in our research, owing to the potential expansion of the cyber threat landscape resulting from the network design implementation. In summary,

our research aims to address the security and reliability challenges faced by modern UAV systems. Our goal is to make a substantial contribution to the development of a resilient and dependable communication network for UAVs by focusing on secure design and the use of trusted components [15].

The proposed threat model for UAVs addresses evolving cybersecurity challenges by integrating a trust-layered approach into threat modeling. Unlike traditional methodologies, xT-STRIDE extends STRIDE by incorporating computational trust, allowing for a more realistic assessment of threats based on trust levels in the UAV system's environment. This approach reduces false assumptions, strengthens security, lowers design costs, and enhances resilience. By adapting STRIDE to account for varying trust levels, xT-STRIDE offers a more accurate and effective threat analysis, ensuring robust protection for UAV systems.

This research presents the xT-STRIDE methodology for threat analysis and risk assessment in UAV cybersecurity. The following section reviews related works and previous studies that have explored threat modeling methodologies for UAV systems, with a focus on ensuring the security of multi-UAV systems during the early development stages. The third section outlines how the methodology defines trust levels, applies the STRIDE framework for threat analysis at each trust level, and selects an appropriate threat list based on the operating environment's trust level. The xT-STRIDE Application Principles section offers a practical, step-by-step guide for integrating the methodology throughout the UAV development lifecycle. The Theoretical Model section discusses the application of the STRIDE method at different trust levels to generate threat lists based on the system's design. In the Model Effectiveness section, the model's effectiveness is demonstrated by modeling the UAV system at two distinct trust levels, highlighting the reduction in threats as higher trust levels are selected. Finally, the research concludes by summarizing its key contributions, emphasizing the benefits of xT-STRIDE, and outlining potential directions for future research.

2 Related works

This section reviews previous studies on the use of threat modeling in UAV systems. The authors emphasize the importance of using threat modeling methodologies to ensure the security of multi-UAV systems in the early stages of development [16]. One such methodology, known as "threat trees," is used to analyze and enumerate the threats that affect the architecture of the Internet of Drones (IoD). The research applies this well-established technique to UAV systems. The use of multi-UAV networks is growing, particularly in disaster management and military operations, as highlighted in reference [16]. This growth can be attributed to considerable technological advancements. However, it is essential to recognize that these networks face security challenges because they belong to the ad hoc network category. This study presents a new approach to address these challenges. The proposed threat model aims to systematically identify and assess risks by considering factors such as adversary capabilities, environmental conditions, targeted entities, and vulnerable domains. What sets this study apart is its focus on the absence of trust competence. Another study builds the design of UAVs on the general principles of the STRIDE framework [17]. Additionally, the design of a secure communication protocol for UAV systems is also considered [18]. Reference [19] presents a proposed solution for an intrusion detection system (IDS), while threat analysis has been applied in the context of safe firmware upgrades [20] and security testing [21]. However, previous studies have not integrated trusted computing layers into their research.

The cybersecurity of UAVs is strengthened by several important studies across various disciplines. Kim et al. propose a mesh-network-based guidance strategy for fixed-wing UAVs, allowing them to achieve and maintain circular formation with high accuracy [22]. Jahangeer and Aldabbas introduce a hybrid cat-swarm optimization algorithm to facilitate seamless vertical handoffs in vehicular ad hoc networks (VANETs) [23]. Naskath develops a fast, multicriteria network selection scheme that combines cat swarm optimization and TOPSIS for optimal handover in VANETs [24]. Balaji et al. design a GAN-based hybrid deep learning model to improve intrusion detection in IoT networks, enhancing overall performance [25]. Sonny et al. present a modified PSO algorithm for efficient autonomous UAV path planning in UAV-assisted wireless networks [26]. Naskath et al. propose a secure connectivity protocol based on game theory to mitigate malicious activities in VANETs [27].

The key components of the communication and operating systems in the UAV framework can be classified into seven primary categories, which align with the xT-STRIDE design methodology. These components—operating systems, sensors, actuators, storage, computing, communication, and applications—are shown in Fig. 1. Communication plays a crucial role in the UAV system network, enabling UAVs to interact with one another as well as with networking backbones and infrastructures to perform various tasks and services [28]. Detailed information on these system elements has been explored in several previous studies [29–36].

Recent studies have focused on cybersecurity in UAVs. Marchetti et al. highlight key cybersecurity challenges in drones and review existing testing methodologies, identifying gaps and suggesting future directions for UAV security research [37]. Kumar et al. provide a comprehensive survey of cybersecurity vulnerabilities in UAV systems and propose countermeasures to enhance their security [38].

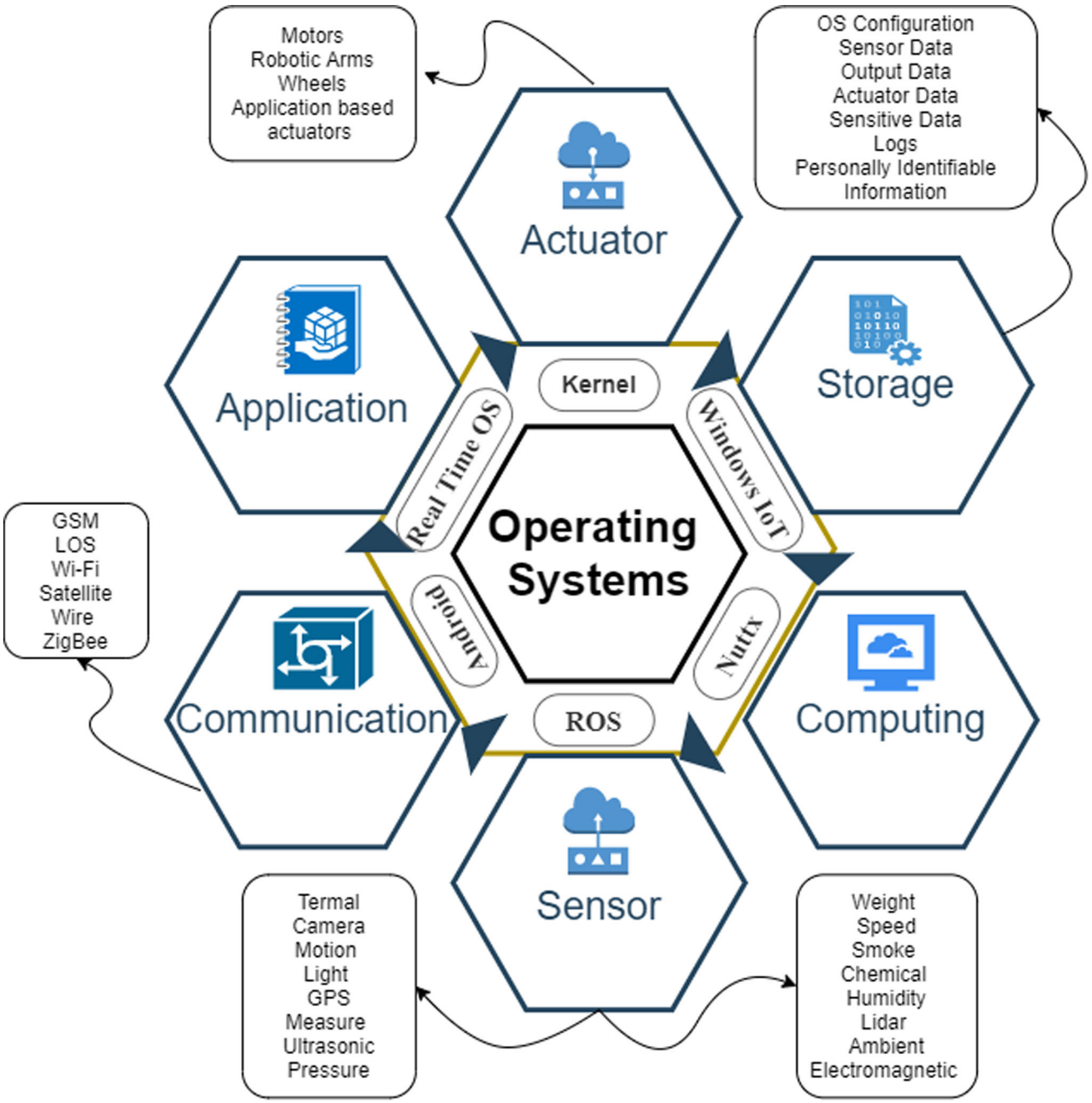

Fig. 1 UAV communication and embedded system architecture

Miao et al. introduce a deep-meta-heuristic system for intrusion detection in UAV networks, improving both detection accuracy and adaptability [39]. Liu et al. propose a multi-component system-based maintenance policy for fixed-wing UAVs, demonstrating its effectiveness in improving reliability through a case study [40]. A TransReSE (Transformer-ResNeXt-SE)-based approach is developed to assess security in UAV swarm networks, enhancing situational awareness and threat detection [41]. Chandran and Kizheppatt design a lightweight mutual authentication protocol based on physical unclonable functions (PUFs) for secure communication in multi-UAV networks [42]. Yang et al. introduce a fault-tolerant and efficient authentication scheme that effectively balances security and performance in manned–unmanned teaming environments [43]. Zhou et al. offer an in-depth survey of AI applications in UAV-enabled wireless networks, highlighting the role of AI in enhancing network performance and management [44]. Li et al. combine blockchain and deep learning to create a secure and efficient defense

strategy for UAV networks, addressing critical privacy and security issues [45]. Akram et al. propose a blockchain-based, privacy-preserving authentication protocol to protect UAV networks from unauthorized access [46]. Zhao et al. present a UAV mobile edge computing framework using Soft Actor-Critical (SAC) to reduce energy consumption while ensuring secure data transmission [47]. These studies underscore the importance of innovative solutions in improving UAV cybersecurity.

There are several standards, models, and frameworks for secure system development, including the Microsoft Secure Software Development Lifecycle (SDLC), OWASP Open Software Assurance Maturity Model (OpenSAMM), Common Criteria (CC), Secure Capability Maturity Model (CMM), and System Security Engineering Capability Maturity Model (SSE-CMM) [48, 49]. However, there is no universal solution for developing a secure system. These models, standards, and frameworks provide guidance on secure design principles, threat modeling, configuration management, risk analysis, secure coding practices, developer training, source code analysis, penetration testing, cryptography standards, security verification, patch management, and incident response. It is essential to create a comprehensive, cohesive model, framework, and set of standards to effectively design and implement a secure system [50, 51]. Owing to the critical nature of system security, the Common Criteria framework has been adopted as a reference model for security assessment, with a particular focus on integrating trusted components.

The components of the CC secure system development approach are shown in Fig. 2 [11]. The Common Criteria (CC-ISO 15408) framework is a comprehensive system developed to address various security contexts [52–54]. It provides a structured methodology for evaluating and certifying the security features and assurance levels of IT systems, ensuring they meet defined security requirements across different contexts [51, 54].

Dronecrypt is an effective threat mitigator in the proposed classification-based UAV system design, addressing spoofing, repudiation, and information disclosure threats [55]. For threats such as denial of service and elevation of privilege, hijacking detection proves to be an effective mitigation strategy [56]. To counter availability threats, a physical security method based on a restart scheme is proposed; however, this method is not universally applicable, though it may be effective in certain scenarios [57]. UAV hacking detection is a promising approach for mitigating availability and denial of service threats [58]. Additionally, physical security solutions can serve as a strong deterrent against tampering-related threats [59]. A novel study suggests potential solutions for spoofing, repudiation, and information disclosure based on homomorphic encryption, a key area of security research [60]. Furthermore, by leveraging blockchain technology, a solution is proposed that incorporates the monitoring capabilities of UAVs [61]. However, the proposed solution for UAV system design is not comprehensive in terms of all design approaches that have been discussed in artificial intelligence research [62].

A previous academic study highlighted the complexity of cybersecurity concerns related to UAVs [63]. The study focused on the growing market for civilian drones, which are increasingly being used in a variety of applications in daily life. However, their heavy reliance on cyber capabilities introduces considerable risks to both individuals and property. The study explored the security, privacy, and safety implications of using civilian drones, identifying both physical and cyber threats, as well as the required security measures. Despite the Federal Aviation Administration's NextGen project, which aims to integrate drones into the national airspace, regulations remain under development. Based on the study, a risk-based classification method for optimizing design effectiveness is proposed. Additionally, a comprehensive security analysis of UAVs is performed, emphasizing the complex nature of cybersecurity issues in UAV systems [64]. The paper examines the increasing use of drones for malicious cyber activities and emphasizes the urgent need for effective countermeasures. It discusses the versatile applications of drones and their growing popularity owing to their ability to provide a bird's-eye view. The paper underscores the high likelihood and frequency of such attacks, their potentially harmful impact, and the necessity for detection, protection, and prevention strategies. Various malicious uses of drones are reviewed, along with potential vulnerability detection techniques. A survey performed in [65] explored emerging threats associated with the use of drones in cyberattacks and emphasized the importance of secure design in smart farming, where UAVs play a pivotal role. While previous work in [66] introduced a game-theoretic model for attack and defense in UAV systems, our study presents a more practical threat-based model, tailored for real-world applicability. Cybersecurity in UAV systems has become increasingly crucial, particularly with the growth of cloud-based solutions and the integration of UAVs at the edge, as highlighted in [67]. Building on this, our study proposes a threat-based design system that aligns with the scenarios discussed in the existing literature [67]. Additionally, a comprehensive study presented in [68] covers all cybersecurity requirements essential for selecting an appropriate UAV system. The primary focus of this study is the Industrial Internet of Things (IIoT). The research emphasizes that accurate drone operations and continuous monitoring are crucial for ensuring the effectiveness of these network configurations. It also highlights that previous studies have not sufficiently addressed the behavior and vulnerabilities of IIoT-enabled drones, statewise verification, or the identification of anomalous drones

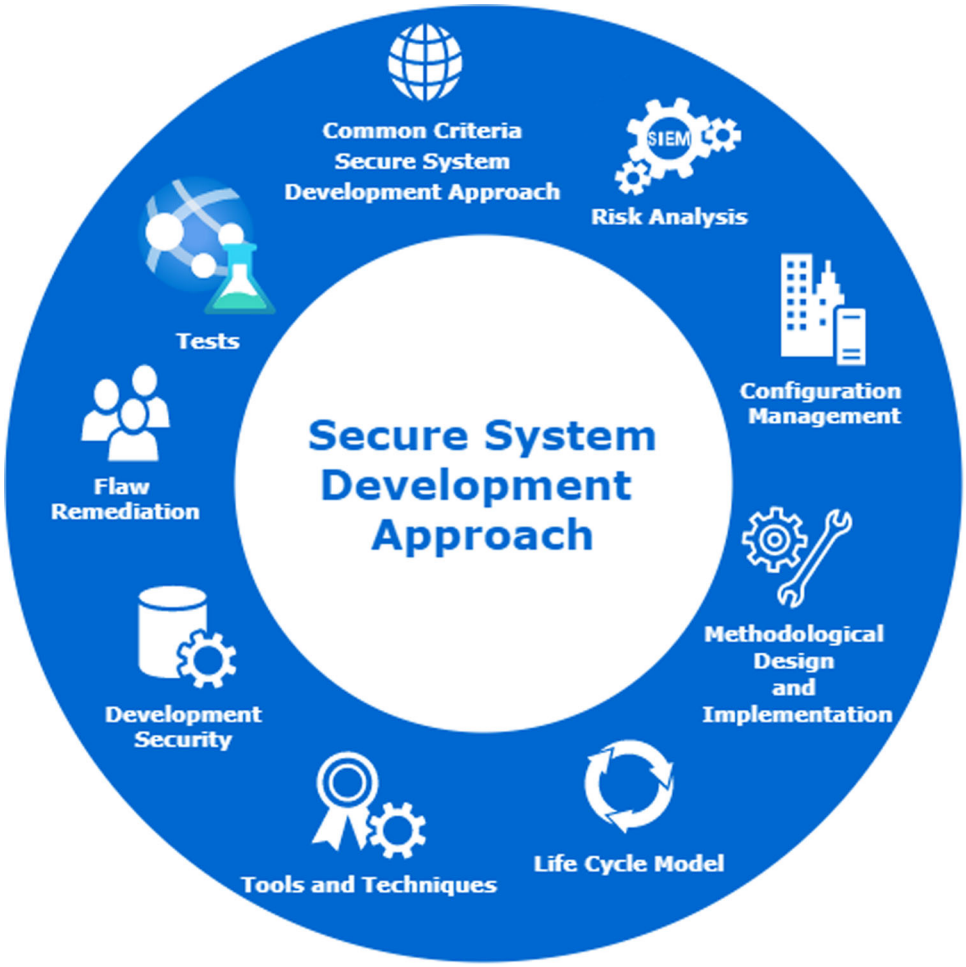

Fig. 2 Secure system development approach

based on different properties. The study concludes that previous solutions fall short in providing the necessary capabilities to address these issues. While implementing such a solution would be ideal, it is impractical owing to its complexity and heavy reliance on the operational aspects of the design rather than the development phase. The paper's trust-based model, based on risk assessment, offers a cost-effective solution that is worth considering [68]. Additionally, a comprehensive survey of potential security issues in Cyber Physical Systems (CPSs) and the Internet of Cyber Physical Things (IoCPT) is presented. The study specifically focuses on CPSs that involve interconnected systems interacting with real-world objects and processes. While related to the IoT, CPS specifically focus on physical, networking, and computational processes. The integration of CPS with IoT has led to the IoCPT. The evolution of CPS is transforming people's daily lives, enabling a broader range of services and applications. However, linking the cyber and physical worlds introduces new security challenges. The study provides an overview of CPS aspects, applications, technologies, and standards, and reviews the security vulnerabilities, threats, and attacks associated with them [69].

3 xT-STRIDE architecture

Risk and threat analysis is a crucial component of any security strategy, offering a framework to identify, understand, and manage potential threats. As cyber threats become increasingly complex, effective risk and threat analysis methods are essential for maintaining system security.

UAVs face numerous cybersecurity challenges, including:

Communication interception and jamming,
Unauthorized access and hijacking,
Data privacy and theft,
Software vulnerabilities,
Supply chain security,
Physical security threats and device tampering,
Risks associated with autonomy and AI integration,
Denial of Service (DoS) attacks,
Insider threats,
Legal and regulatory compliance issues [70].

Given the considerable cybersecurity challenges, providing cybersecurity for UAVs is a complex task. A common

approach to addressing these challenges is by incorporating trusted components, where cybersecurity-related vulnerabilities are eliminated and certified [54]. In this context, a threat analysis and risk assessment approach is used to model the cybersecurity posture of UAV design. xT-STRIDE is designed to be a tool for cybersecurity engineers, considering all threat scenarios and integrating trusted component utilization into the threat analysis process.

Because xT-STRIDE allows for customizable trust levels based on any asset categorization method, it can also model topological challenges associated with drone swarms. In particular, the study's example models five security levels, with the highest being application-level trust, which also addresses vulnerabilities in drone swarm applications [71].

The field of risk and threat analysis has advanced considerably over the years, leading to the development of various approaches. These approaches can generally be classified into two main categories: quantitative and qualitative. Quantitative methods, such as the Risk Analysis Methodology developed by the US Department of Defense, rely on numerical calculations and data-driven insights. In contrast, qualitative methods, such as the Threat Analysis Method (TAM) developed by the US Department of Homeland Security, emphasize policy and expert judgment. Additionally, there are specialized risk and threat analysis methods for aerial systems, such as Coras [72] and Mehari [73]. Recent research on risk assessment has used asset category classification and employed threat modeling for each category, as seen in the SARA framework [74]. However, this study differentiates itself by focusing on the engineering design of individual assets. By adopting an engineering approach that aligns with CC standards, xT-STRIDE uses trust levels to develop security-enhanced assets.

The primary goal of risk and threat analysis is to identify and prioritize potential threats, followed by the development of strategies to mitigate or eliminate them. To achieve this, it is essential to have a thorough understanding of the organization's or system's risk profile. This includes understanding the sources of threats, their attack methods, the potential damage they can cause, and the system's level of vulnerability.

In addition to quantitative and qualitative approaches, several other risk and threat analysis methods have been developed over the years. These include the Systematic Risk Analysis Method (SRAM) developed by the National Institute of Standards and Technology (NIST) [75], the Risk and Threat Analysis Framework (RTAF) developed by the Security Risk Management Institute (SRMI) [76], and the Risk Management Framework (RMF) developed by the International Organization for Standardization (ISO) [77].

Each of these approaches has its own strengths and weaknesses, and the most effective method for a given organization depends on its specific risk and threat profile. For instance, the SRAM approach is particularly well-suited for systems with complex security requirements, whereas the threat analysis method is more appropriate for systems with simpler security requirements [78, 79].

xT-STRIDE is built upon the STRIDE framework [10] for two main reasons. First, STRIDE models threats in a way that closely aligns with the motivations of attackers, allowing for more effective identification and mitigation of security risks by starting the threat modeling process near the source of potential threats. Second, STRIDE is straightforward to understand and apply, whereas other approaches often require more platform-specific expertise [80–84].

Risk and threat analysis are essential components of any effective security strategy, and over time, various methodologies have been developed. However, a considerable limitation of existing models is the lack of integration of a computational trust layer. In this paper, we introduce xT-STRIDE, a framework that integrates computational trust into threat analysis, specifically designed for UAV systems. UAVs face unique security challenges owing to factors such as cost, application domain, data security requirements, battery life, and performance constraints, all of which require customized security solutions. Additionally, a comprehensive risk assessment is crucial in defining the appropriate security levels. Typically, data classification models are used to determine these requirements, with military systems often adopting a five-level classification approach, including: unclassified, sensitive, confidential, secret, and top secret.

This classification strategy is based on the confidentiality, integrity, and availability of the associated document, system, or data. The security provisions for the UAV system encompass broader constituent types that must be considered beyond the typical security considerations. Because a UAV system consists of multiple interconnected subsystems, each acting as a network, standards such as ISO 27001 and ISO 15408 may be relevant for security standardization. However, the unique characteristics of UAV systems—such as mobile networks that continuously change locations and the strict connections between land and air components—require a security design tailored to address these specific features. Meanwhile, not all UAV systems require the same set of security provisions that satisfy user needs. Also, some security considerations can be just basic security implementation because of the definition of the UAV system application.

The STRIDE model is a threat modeling technique used to identify potential threats against the system under investigation. However, existing STRIDE implementations do not account for trust levels when designing a system. A similar study applies STRIDE to CPSs but overlooks trust level classification [85]. There is also a more automated approach available that does not incorporate trust level classification [86]. STRIDE has been applied to 5G core slicing without considering trust levels [87], and multiple threat modeling

frameworks in a single tool have been considered without addressing trust level integration [88]. Applying trust level classification to the STRIDE model creates a more accurate threat list by eliminating false assumptions about the operational environment. In an untrustworthy environment, STRIDE would identify more threats by removing pre-established trust assumptions. Conversely, in a trusted environment, xT-STRIDE would result in fewer identified threats owing to the use of trusted components.

The novel xT-STRIDE method works as follows:

The algorithm begins by defining x distinct trust levels.
For each trust level, the STRIDE method is applied to the system under design to generate a corresponding threat list.
To provide security, the appropriate trust level for the operating environment is selected, and the threat list aligned with that trust level is produced as the final output.

Section 2 presents the design details of UAVs. The collected information on related work can be summarized across various engineering design levels, ranging from high-level to low-level design stages: System Level, Interface Level, Application Level, Operating System Level, and Hardware Level. These five levels of design follow a general system design approach, progressing from high-level to low-level design. Because UAVs are categorized as system design projects, this five-level structure is well-suited. To adapt xT-STRIDE to UAV design, we choose to follow this five-level design approach and modify xT-STRIDE to create 5 T-STRIDE.

To address the challenges of UAV design, the solution proposed in this study uses the 5 T-STRIDE framework to enhance the security of UAV systems. The adoption of the 5 T-STRIDE approach allows for the simultaneous consideration of both design complexity and system architecture.

Trust levels can vary in granularity depending on the application. For example, the interface level can be further subdivided into more specific layers, such as the network layer, VLAN layer, and tunnel layer, for a more detailed analysis. The UAV system is structured into five distinct layers based on varying component usage scenarios, and STRIDE threat analyses are performed accordingly. Components such as hardware, operating systems, and sensors may be obtained from different vendors, and security assessments can be more effectively performed by considering the security assurances provided by these components. Moreover, the 5 T-STRIDE model enables precise security analysis based on the desired level of trust. Considering the secure UAV system as depicted, five trust levels can be defined as follows:

System Level: At this level, the UAV system is viewed as a unified entity, encompassing UAVs, ground centers, and personnel. The security perimeter is defined around this entire system, meaning that components inside the perimeter are trusted, while threats are expected from outside the perimeter.
Interface Level: Here, the security perimeter is established around the major components of the UAV system. Threats are expected to arise from the interfaces between these key components. For example, a UAV may face threats from its physical interfaces, but all internal components are trusted.
Application Level: At the application level, threats are expected to affect the applications running on the UAV and other associated components. Operating systems and hardware components are considered to be trusted in this level of design.
Operating System Level: At this level, potential threats are mainly associated with the operating systems and higher-level components. Hardware components remain trusted at this stage.
Hardware Level: At this level, the highest level of security is considered. Any component can be attacked, including hardware components.

The security requirements of a UAV can vary depending on its intended application. For example, when acquiring a UAV for military purposes, a thorough security analysis must be performed across all system levels. In contrast, for recreational use, system-level security measures may be sufficient. For commercial applications, focusing on application-level security measures is often adequate.

To generate the threat list for each trust level, the STRIDE model is used as the threat analysis framework for the security classification. Developed by Microsoft, STRIDE is a model designed to identify computer security threats, offering a mnemonic for six categories of security threats:

Spoofing
Tampering
Repudiation
Information Disclosure
Denial of Service
Elevation of privilege

Each threat category corresponds to a breach of a specific security property. Spoofing targets the authenticity property, tampering affects integrity, repudiation undermines the non-repudiation property, information disclosure violates confidentiality, denial of service impacts availability, and elevation of privilege compromises authorization security properties.

Although xT-STRIDE has been specifically applied to unmanned aerial vehicle security (UAVSEC), its applicability extends to any system wherein the STRIDE framework is relevant. This versatility stems from the ability to categorize each computational system into various organizational levels,

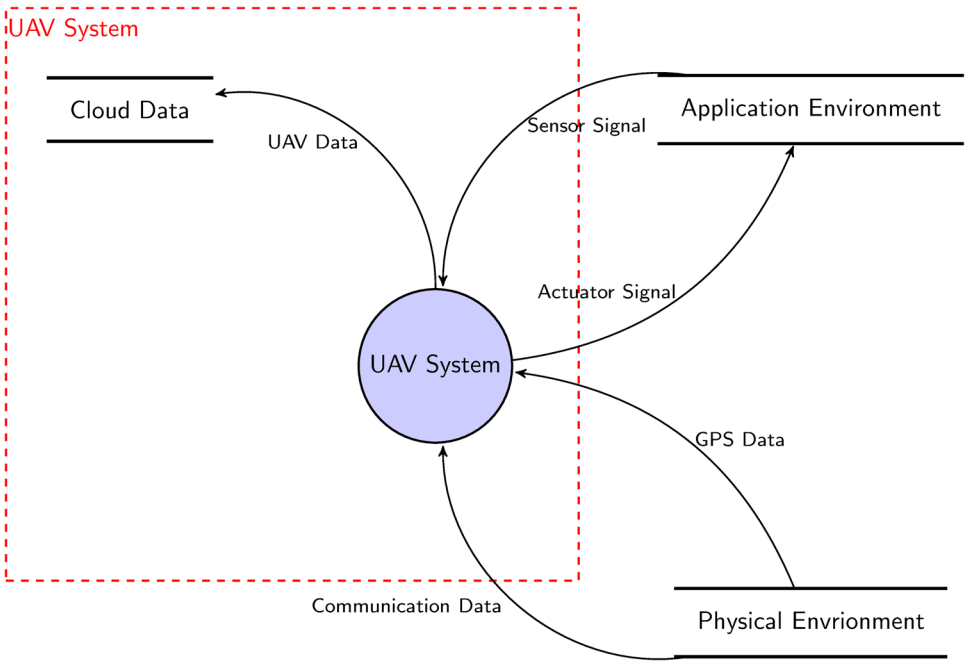

Fig. 3 System-level model

allowing xT-STRIDE to be applied across multiple levels of an organization. For this analysis, five distinct topological UAV system models corresponding to the defined 5 T-STRIDE levels have been developed. The Microsoft Threat Modeling Tool has been used to implement the STRIDE model for each trust level. Figures 3, 4, 5, 6, 7 show the models created using the Microsoft Threat Modeling Tool, with each figure generated from text-based level definitions and LaTeX compilation. Figure 3 shows the System Level Model, which aligns with the system-level trust assumption. This model indicates that the components in the system are trusted, while threats are identified at the interfaces of the overall system.

4 xT-STRIDE application principles

In applying the 5 T-STRIDE framework to UAV security, the next step involves generating five levels of STRIDE-based threat models for UAVs. After creating these five threat models, the cybersecurity decision is to assign a trust level to one of the five levels. By assigning a trust level, all components below that level are considered trusted. This trust level assignment requires selecting components that are sufficiently established as trusted. In related works, the Common Criteria approach is referenced to provide such trust guarantees through an industry-accepted methodology. At the current technology level, this approach is a valid assumption because all necessary technologies to implement xT-STRIDE are available, as demonstrated in related works. Figures 3–7 show the process of generating the five-level threat models for UAV design.

The effectiveness of xT-STRIDE is demonstrated by modeling the UAV system with two distinct trust levels. The first level is hardware-level trust modeling, which includes all hardware and the above layers that need to be secured. In this level, no components are assumed to be trusted. The second level is operating system-level trust modeling, where hardware is selected from trusted components, and only the software and higher layers of the UAV require security. The two corresponding threat models are shown in Figs. 6 and 7. Figure 8 shows the total number of threats that result for each security level once trust is established at that level. Establishing a trust level at a particular layer implies that all components in the layers below it are considered trusted. Because trust level categories can be customized for different scenarios, trust levels can also account for the topological placement of UAVs. For instance, UAVs in a specific geographical region can be assigned a trust level, meaning UAVs designed for that region are trusted and excluded from the threat model. These trusted UAVs are certified to eliminate potential vulnerabilities. Figure 9 shows the resulting threat items from Fig. 8, where threats are categorized into individual STRIDE categories.

The interface-level trusted model is shown in Fig. 4. In accordance with the definition, the system components are secured at the highest level of system design. In this model, threats in the components are not considered because the subcomponents are trusted.

Figure 5 shows the application-level trusted model. In this trust level, only the application-level components are consid-

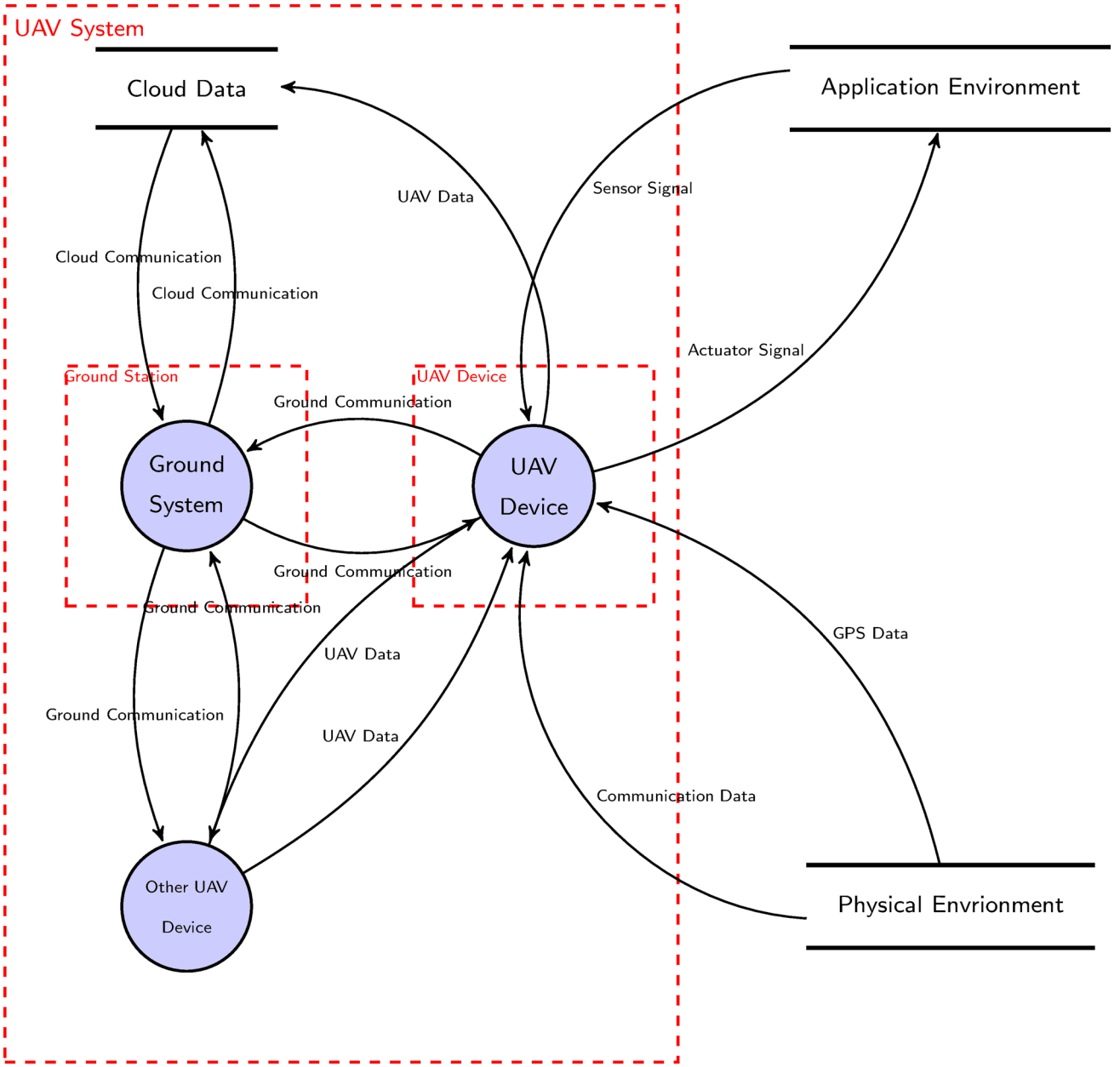

Fig. 4 Interface-level model

ered potential sources of threats, while the operating systems and hardware are classified as trusted components.

The trust assumption is reduced by one level, with threats now considered as potentially originating from the operating systems of the components shown in Fig. 6. At this trust level, the hardware is still regarded as trusted.

Figure 7 shows the highest level of threat occurrence. At this level, all the components in the system, both software and hardware, are considered potential threat sources. Consequently, this level is expected to result in the maximum number of potential threat instances.

The generated five-level threat models need to be evaluated to understand the effort saved using trusted components. The basic approach to assessing this effort-saving is counting the number of threats generated by each model. Because each threat in the models represents excess work required to eliminate it from the design, a decrease in the number of threats generated by higher-level models demonstrates the effort-saving benefit. Once a trust level is selected, the threats associated with that level must be eliminated. This represents a trade-off in design with xT-STRIDE. By selecting a higher trust level, the components at lower levels are also assumed to be trusted, meaning they are already secured. High-level

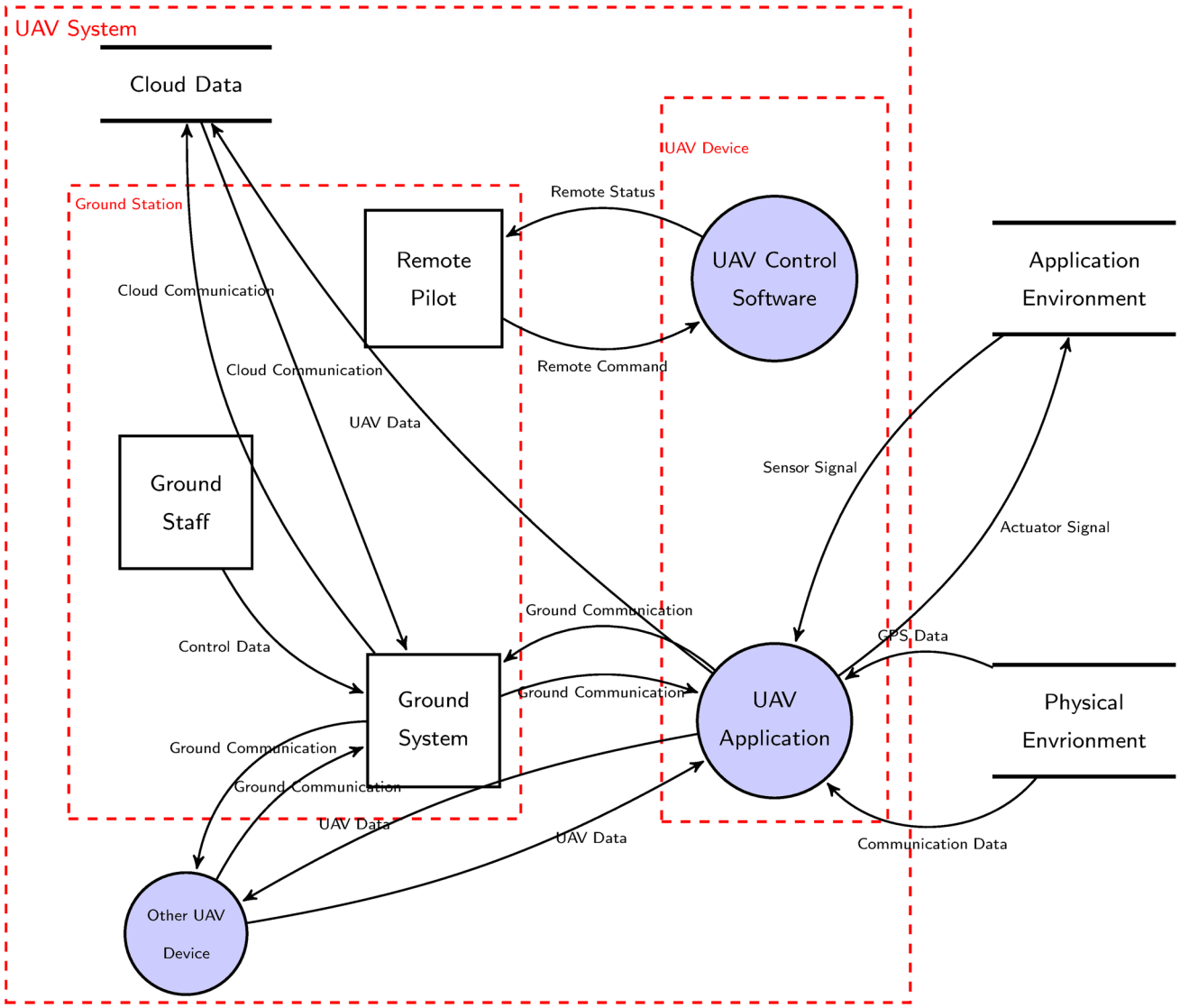

Fig. 5 Application-level model

components are those that need to be secured. In an ideal scenario, it would be optimal to select all components as trusted to maximize security. However, choosing to trust all components would result in very high production costs, which manufacturers typically want to avoid. On the other hand, if no trust level is selected, all components are treated as needing to be secured, which could lead to a less secure design because securing all components in such an environment may be unfeasible. Essentially, selecting the appropriate trust level in xT-STRIDE is a trade-off between cybersecurity and production costs. Figures 8 and 9 can be used to guide decisions in UAV design by thoroughly analyzing the threat posture at each model level.

The analysis produced threat counts, as shown in Fig. 8, for the traffic monitoring scenario example. Figure 9 shows these threat counts according to each threat category. Based on the data presented in both figures, it has been observed that the attack paths in the UAV system design have been considerably reduced. This reduction can be attributed to the decrease in the number of potential threats with each incremental increase in the trust level, from hardware to the system level. These findings suggest that increasing the system's trust level can greatly reduce the overall attack paths, thereby enhancing the security of the UAV system. This case study demonstrates the application of the xT-STRIDE framework, incorporating five distinct levels of trust. While other methodologies may vary in their layer modeling and the prioritization of each layer, our approach consistently focuses on eliminating the components in the trusted layer that fall under the scope of risk assessment. Additionally, each trust level builds upon the previous one, creating a comprehensive and layered system of trust.

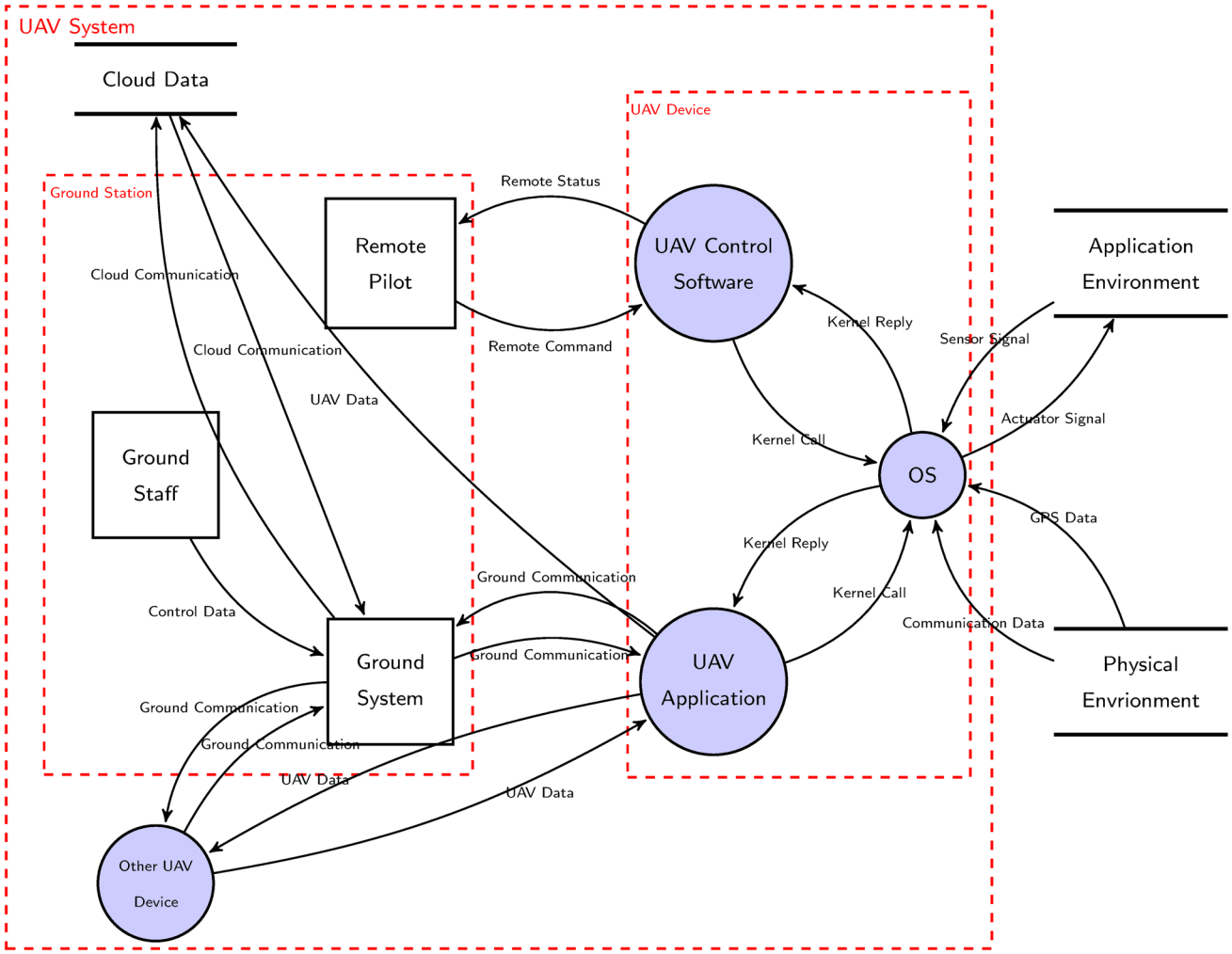

Fig. 6 Operating-system-level model

As shown in Figs. 8 and 9, the overall trend suggests that reducing complexity in 5 T-STRIDE leads to a decrease in threat counts. However, an exception occurs in the Elevation of Privilege threat category, where hardware considerations result in fewer threats. This phenomenon may be attributed to the increased complexity that hinders elevation of privilege attacks.

5 Theoretical model

The main goal of this modeling is to improve the overall cybersecurity risk analysis process through the application of mathematical modeling techniques [89, 90]. Our approach models each cyberattack, starting from system-level access and progressing to the hardware level, ultimately leading to the attacker's objective. At each level of the attack, we can observe varying degrees of security. The attack process is shown in Fig. 10.

The effectiveness of xT-STRIDE in the context of UAV system security is demonstrated through a theoretical approach, which assumes that the vulnerability factor for each trust level can be exploited. This approach is well-suited for the UAV system design.

The xT-STRIDE method targets the UAV development process, where all cybersecurity requirements are incorporated using a secure-by-design approach [91]. With this approach, vulnerabilities are addressed and eliminated during the design phase of the development process. Without loss of generality, the total work involved in the development process is the sum of the individual vulnerability items, which is referred to as total vulnerability. This total vulnerability reflects the overall effort required to secure the system. Equation 1 defines the calculation for this. A vulnerable system can lead to vulnerability propagation, adding residual vulnerabilities to other systems [91]. D[i,j] represents the residual vulnerability, where a vulnerable system i introduces residual vulnerability to system j. This residual vulnerability model

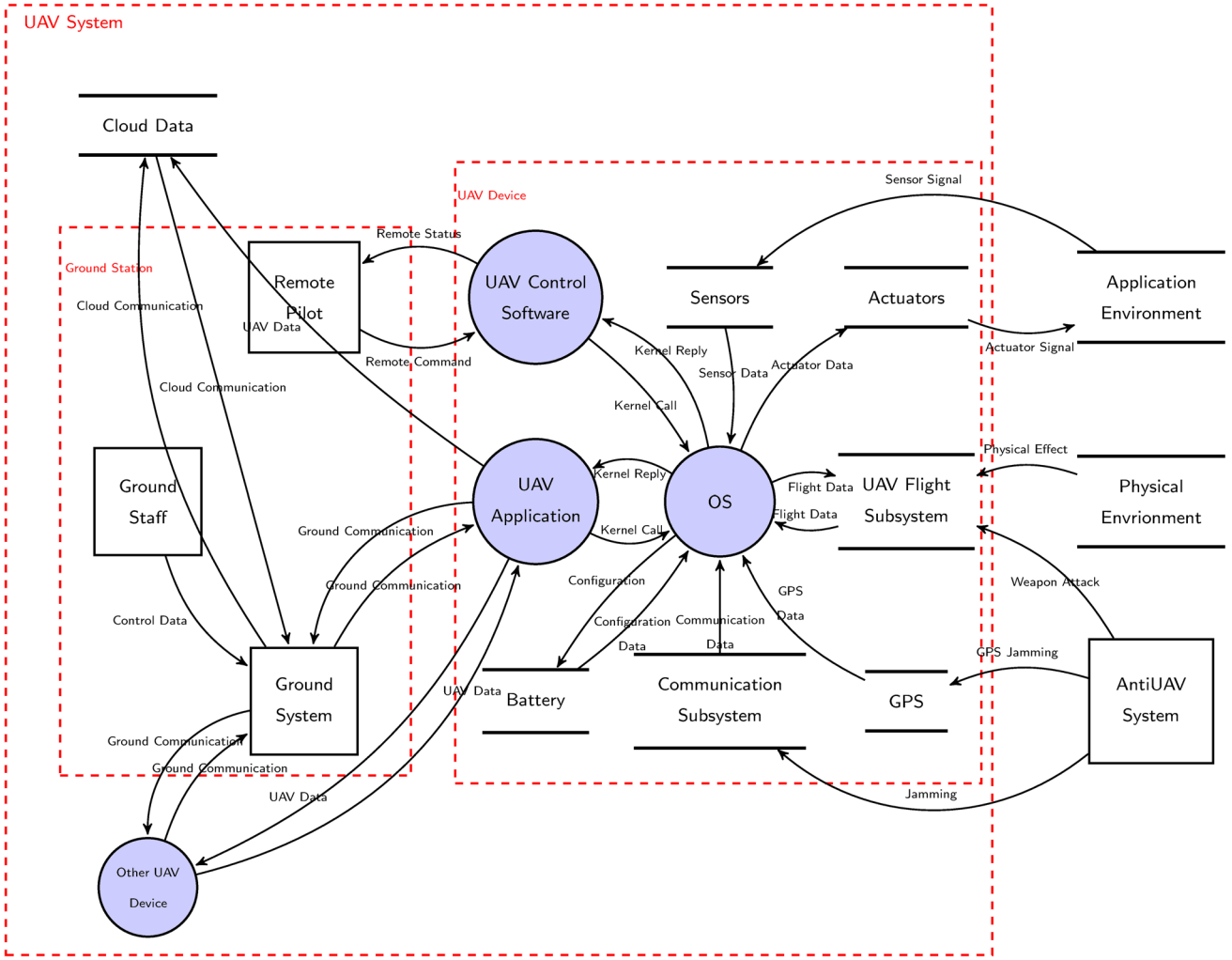

Fig. 7 Hardware-level model

is incorporated into the computation of total vulnerability in Eq. 1. Equation 2 provides a simplified matrix form of all the individual vulnerability values used in Eq. 1. Equation 5 is derived from the principle that the residual vulnerability of a system will always be less than the system's original vulnerability [91]. To model this reduced residual vulnerability, we introduce a spread factor, which is less than 1. In the examples, we have chosen a conservative value of 0.2 for this spread factor. In actual system design, the spread factor can be chosen by running statistical tests for the attack tree analysis of the real threats.

The analysis was performed using 5 T-STRIDE and five distinct trust levels, as outlined in Sect. 4 and presented in Table 1. If a level is not trusted, it may introduce vulnerabilities to the other levels; however, if the level is trusted, the assumption is that no vulnerability injection occurs. This approach aims to demonstrate the effectiveness of xT-STRIDE in improving UAV system security.

To enhance the effectiveness of our methodology, we introduce a parameter called "level vulnerability." This parameter is obtained by assessing the probability of a successful attack on a vulnerability at a given level. To ensure broad applicability, this value can be determined by aggregating the maximum function in cases where multiple vulnerabilities are present. Alternative aggregation methods may also be employed, if they do not compromise system integrity. Additionally, the computation of this parameter incorporates a dependency factor, reflecting the interconnected nature of the computational levels in UAV systems. In the context of trust level modeling, we define trust layers that possess zero inherent vulnerability and zero injected vulnerability, due to the comprehensive assessment of components at each trust level. For level i, the overall vulnerability is given by Eq. (1), where v_i represents the base vulnerability of level i, $D(j, i)$ denotes the vulnerability introduced into level i from other levels, and $D(i, j)$ represents the vulnerability

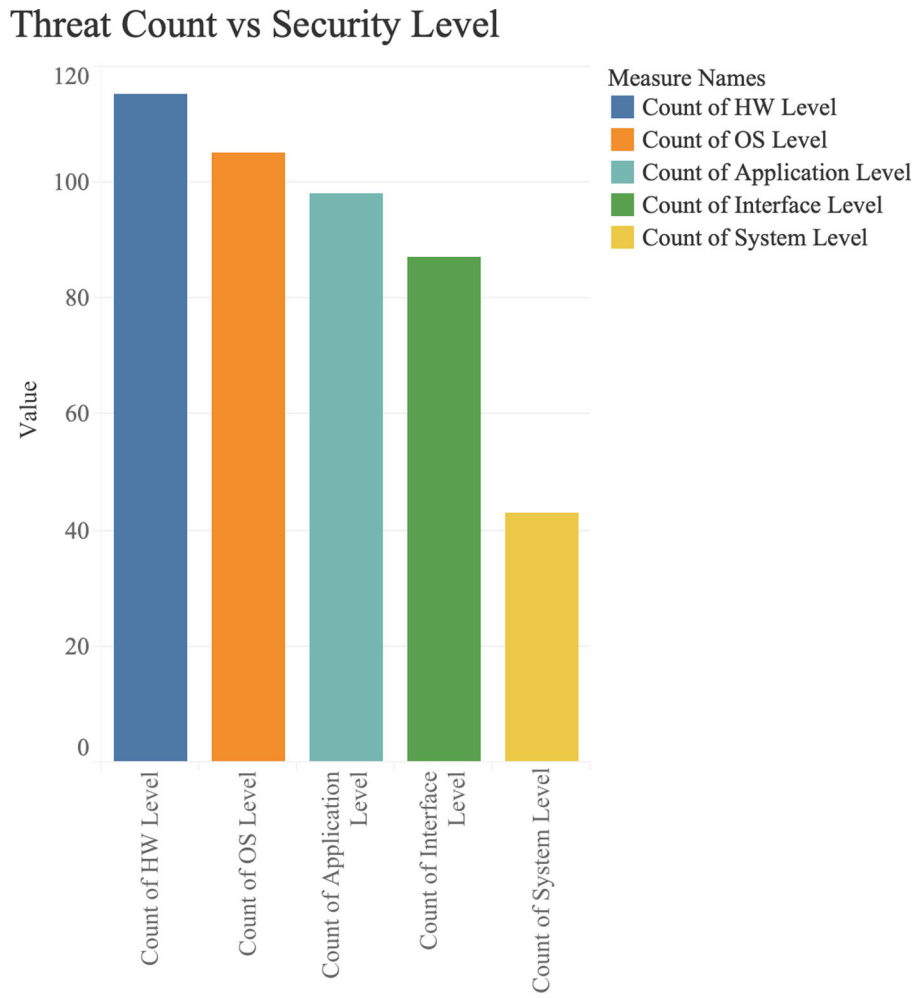

Fig. 8 Threat counts for each security level

Count of Application Level, count of HW Level, count of Interface Level, count of OS Level and count of System Level. Color shows details about count of Application Level, count of HW Level, count of Interface Level, count of OS Level and count of System Level.

propagated from level i to other levels.

$$V_i = v_i + \sum_{j \neq i} D(j, i) + \sum_{j \neq i} D(i, j) \tag{1}$$

If the level is trusted, then v_i and $D(k, j) for k \in [0, i]$ will be zero. The general structure is represented in matrix form in Eq. (2).

$$\begin{matrix} v_0 & D(0,1) & D(0,2) & D(0,3) & D(0,4) \\ D(1,0) & v_1 & D(1,2) & D(1,3) & D(1,4) \\ D(2,0) & D(2,1) & v_2 & D(2,3) & D(2,4) \\ D(3,0) & D(3,1) & D(3,2) & v_3 & D(3,4) \\ D(4,0) & D(4,1) & D(4,2) & D(4,3) & v_4 \end{matrix} \tag{2}$$

Using this matrix definition, the total vulnerability of an operating system can be calculated by summing the elements highlighted in yellow in Eq. (3).

$$\begin{matrix} v_0 & D(0,1) & D(0,2) & D(0,3) & D(0,4) \\ D(1,0) & v_1 & D(1,2) & D(1,3) & D(1,4) \\ D(2,0) & D(2,1) & v_2 & D(2,3) & D(2,4) \\ D(3,0) & D(3,1) & D(3,2) & v_3 & D(3,4) \\ D(4,0) & D(4,1) & D(4,2) & D(4,3) & v_4 \end{matrix} \tag{3}$$

Similarly, the total vulnerability for each level can be calculated using the same approach.

Cybersecurity decisions for design challenges such as UAVs require deterministically guaranteed claims. This necessity arises from the fact that any cyber incident involving UAVs can lead to catastrophic consequences for public safety. To establish such guarantees, analytical approaches

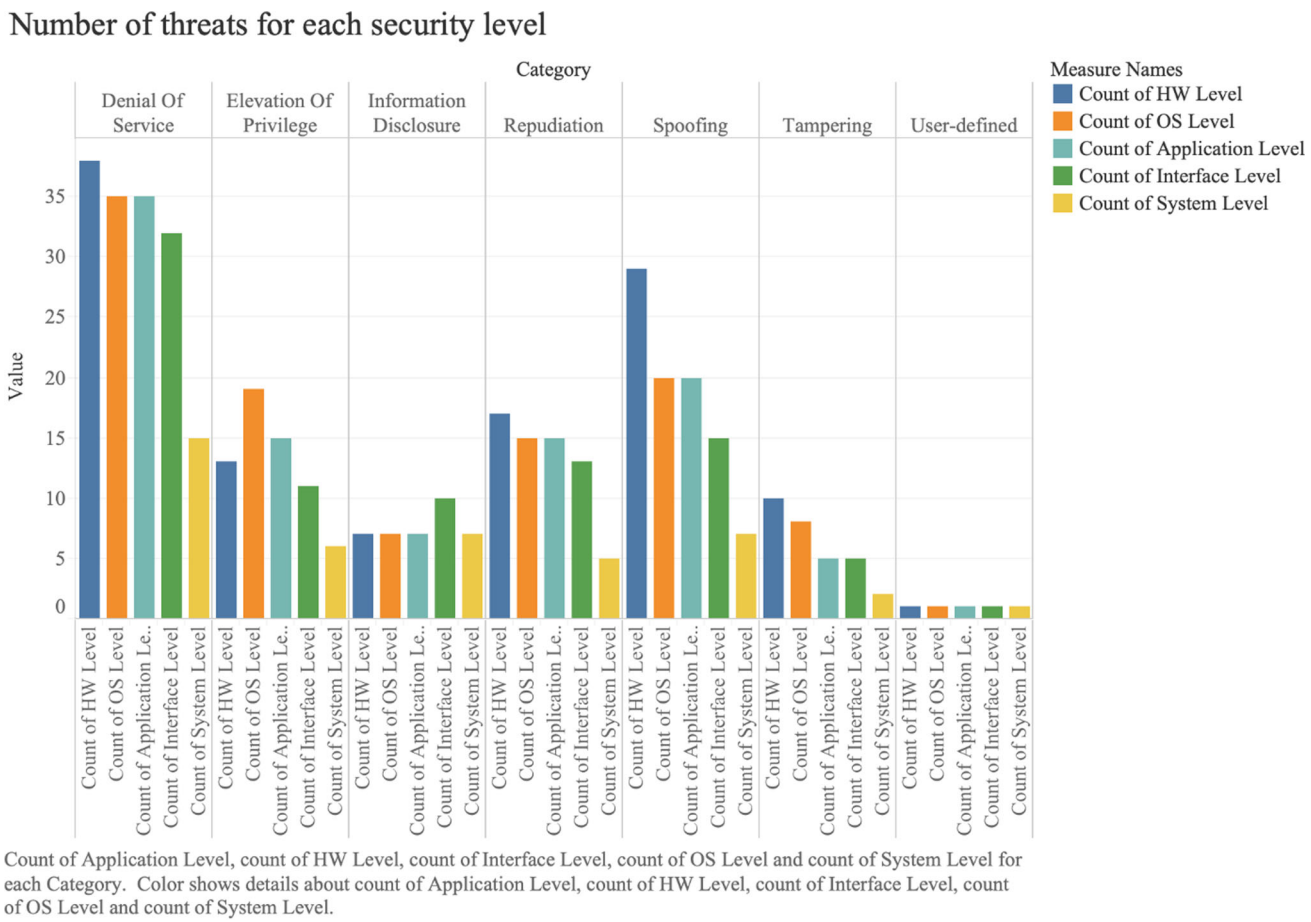

Fig. 9 Threat counts for each security level with respect to threat categories

that produce deterministically proven outcomes are essential. One such approach is Petri Net modeling, which is widely used to represent various public safety technologies. In this study, the Petri Net modeling approach is used to evaluate the performance of xT-STRIDE in delivering deterministically proven cybersecurity claims.

For verification purposes, a Petri Net simulation is implemented to demonstrate the effectiveness of 5 T-STRIDE. The general Petri Net structure used in the modeling is shown in Fig. 11. This simulation is used to evaluate both the cybersecurity performance and the overall effectiveness of 5 T-STRIDE.

In this Petri Net model, an attacker can be represented with varying skill levels. When there is no vulnerability, the attacker's skill is modeled as a normal random variable with a mean of zero and a variance of 1.0. If the attacker's skill level exceeds zero, the attack is considered successful; otherwise, it fails.

To simulate increased attack success, a limited number of vulnerabilities were injected into a level. This vulnerability injection is modeled as a value between 0 and 1, representing the system's vulnerability level. Consequently, at each level, an attack will succeed if it satisfies the conditions defined in Eq. (4).

$$Normal(V_l, 0.5) > 0 \tag{4}$$

Global vulnerability data from the National Vulnerability Database (NVD) is used to demonstrate the effectiveness of the proposed method. To support the analysis, we developed a vulnerability level model ranging from 0 to 1, divided into 10 quantization levels. The vulnerability level of each product is determined based on the number of published exploits in the NVD. Given the minimum quantization step of 0.1, we found that the ARM Cortex-A does not exhibit a vulnerability level above medium, resulting in an assigned value of 0.1. At each level, the most secure technology is assumed to have a baseline vulnerability level of 0.1. The values are determined according to the 2022 reported vulnerabilities in NVD [92].

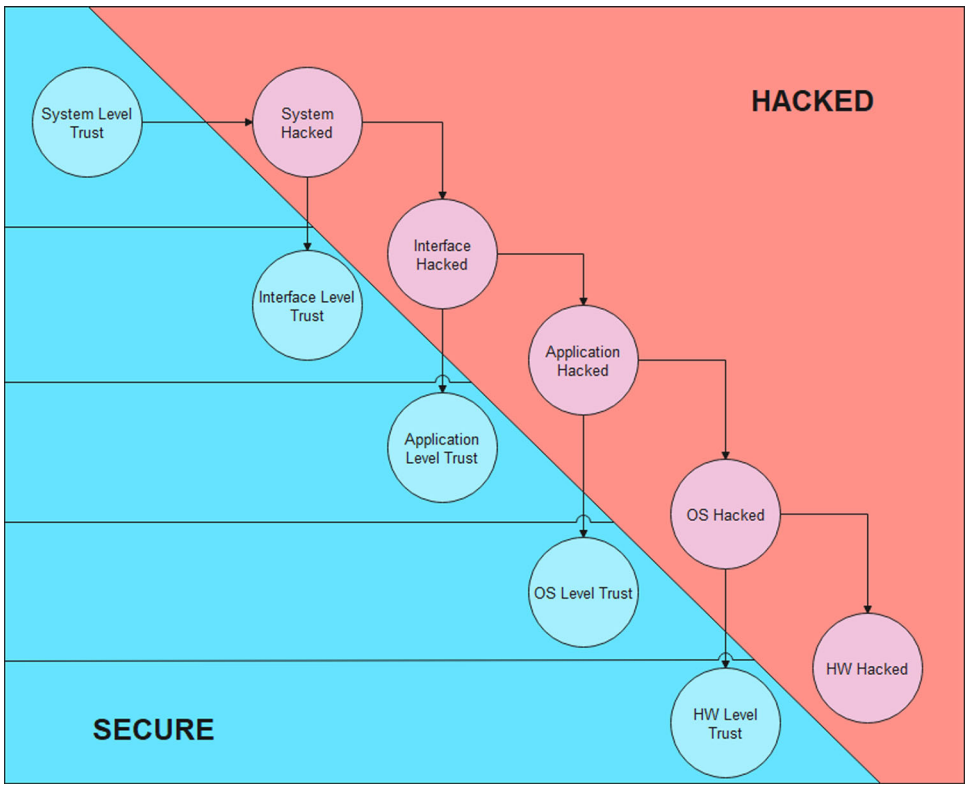

Fig. 10 A typical attack process with trusted components

Table 1 Level indexes

Security Level	Level Index
HW	0
OS	1
Application	2
Interface	3
System	4

6 Model effectiveness

To demonstrate the effectiveness of the theoretical model, it is necessary to assign values to both the base vulnerability and the injected vulnerability. The selection of these values for evaluating the theoretical model is guided by the following two basic principles:

(1) If the ith system is more secure than the jth system,

$$v_i < v_j$$

(2) Because the vulnerability injected by system i into system j is expected to be less than the vulnerability of the ith system, the following inequality should also hold:

$$D(i, j) < v_i$$

In Tables 2, 3, 4, 5, 6, we adhere to the two abovementioned rules to determine a valid value set for theoretical model evaluation.

The approach for calculating the vulnerability of a UAV is based on the scenario depicted in Fig. 12. In this scenario, the attacker's objective is to compromise the UAV system, which requires the exploitation of three vulnerabilities. Therefore, if only two vulnerabilities exist in the system, it can be assumed to remain secure overall, even though a single drone may be compromised. Without loss of generality, and in accordance with this scenario, we consider the presence of two vulnerabilities to be a low-risk condition because the system remains operational. The quantization of vulnerabilities is modeled based on this attack scenario.

The calculation of the vulnerability level tables proceeds as follows:

Each level begins with a base vulnerability value of 0.1, as referenced in [5]. Semi-quantitative vulnerability assessments classify values below 20 out of 100 as low or very low [5]. For simulation purposes, the authors of this study averaged the low-risk values, resulting in a value of 10, which corresponds to 0.1 on a 0–1 scale.
For each additional vulnerability, an extra 0.1 is added to the base vulnerability value.

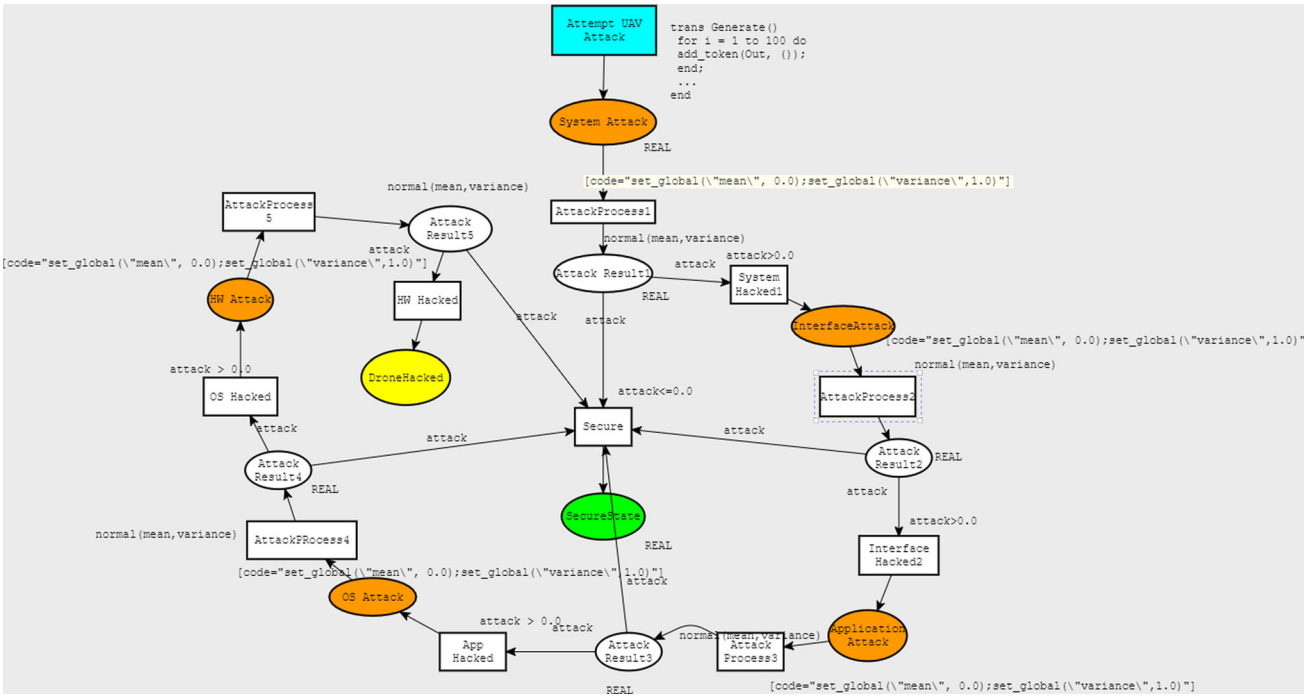

Fig. 11 Petri Net simulation model

Table 2 Hardware vulnerability levels

HW	Vulnerability
ARM Cortex-A	0.1
Intel Processor	0.2
AMD Processor	0.3

Table 3 Operating system vulnerability levels

OS	Vulnerability
Windows 10	0.1
Mac OS	0.2
Linux	0.3

Table 4 Application vulnerability levels

Application	Vulnerability
JVM	0.1
.net Framework	0.2
Docker	0.3

Table 5 Interface vulnerability levels

Interface	Vulnerability
Sophos XG Series	0.1
WatchGuard Firebox T15	0.2
Fortigate Rugged	0.3
iptables	0.4

Table 6 System vulnerability levels

System security	Vulnerability
Perimeter security + Access Control + Surveillance + Alarms	0.1
Perimeter security + Access Control + Surveillance	0.2
Perimeter security + Access Control	0.3
Perimeter security	0.4

To support the analysis, we established a vulnerability level model ranging from 0 to 1, divided into 10 quantization steps. The vulnerability level of each product was determined by calculating the number of published exploits in the NVD database. Given a minimum quantization level of 0.1, we found that the ARM Cortex-A currently has one vulnerability level above medium, resulting in a calculated value of 0.1×1.

By applying these underlying assumptions, we derived varying degrees of vulnerability for each trust level, as

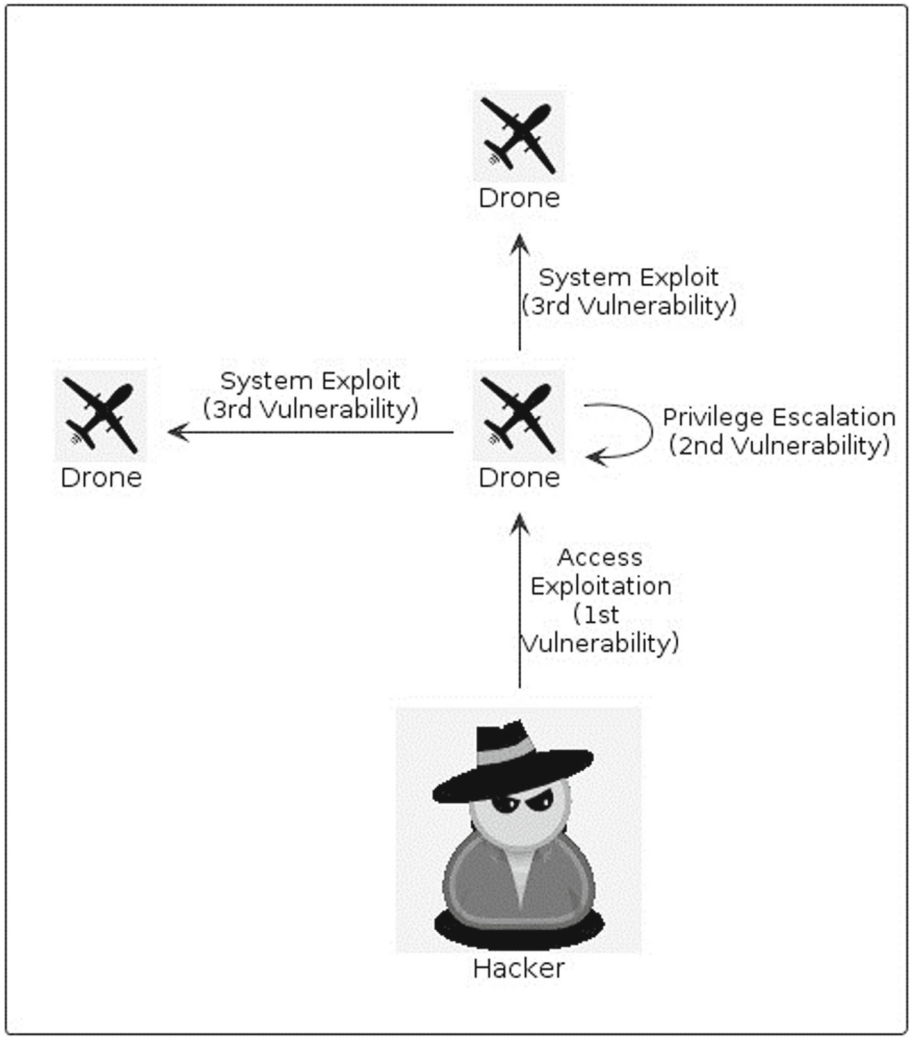

Fig. 12 UAV attack scenario

demonstrated using representative products in Tables 2, 3, 4, 5, 6.

Hardware vulnerabilities are analyzed and specified in Table 2.

OS vulnerability levels are shown in Table 3.

Application vulnerability levels are shown in Table 4.

Interface vulnerability levels are shown in Table 5.

System vulnerability levels are shown in Table 6.

In 2022, 90% of large organizations adopted Puppet to support secure development and DevOps practices, while overall adoption of DevOps techniques reached 80%. To assess the impact of these security practices, we analyzed the vulnerability injection ratio, which reflects the rate of insecure software and product development. For modeling vulnerability injection across all levels, we referenced global secure development practices.

Without loss of generality, we calculated $D(i,\ j)$ as described below, providing a current snapshot of the security posture of these organizations. However, the calculation of $D(i,\ j)$ requires further investigation in a more detailed study owing to the potential for various correlations arising from inter-level dependencies. The authors intend to perform more detailed quantization studies on the computation model presented in this study. To simplify the analysis, we have assumed a uniform distribution of each level's vulnerability across other layers. In the case of a 5-level design, this results in a spread factor of 0.2 in Eq. (5).

$$D(i,\ j) = v_i * 0.2. \tag{5}$$

To evaluate the performance of xT-STRIDE, we compared the most and least vulnerable configurations, as well as an OS-level trust configuration. Without any trust level, the vulnerabilities are as follows $V_{lv} = \{0.26,\ 0.26,\ 0.26,\ 0.26,\ 0.26\}$, $V_{mv} = \{0.78,\ 0.78,\ 0.78,\ 0.78,\ 0.78\}$. With OS level trust,

$V_{lvt} = \{0.06,\ 0.06,\ 0.16,\ 0.16,\ 0.16\}$, $V_{mv} = \{0.18,\ 0.18,\ 0.48,\ 0.48,\ 0.48\}$. The performance improvement measured by 5 T-STRIDE is depicted in Fig. 13.

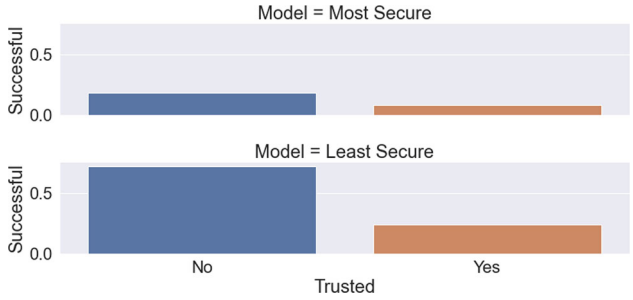

Fig. 13 Petri Net simulation results

The results presented in Fig. 13 show the Petri Net simulation results for the cyber attack scenario depicted in Fig. 12. Two bar charts are shown in Fig. 13. The upper chart represents a system with fewer vulnerabilities, while the lower chart illustrates a system with higher vulnerability values. The horizontal axis in Fig. 13 indicates whether xT-STRIDE was used in the system's cybersecurity design. Bars labeled "No" represent designs that do not use xT-STRIDE, while bars labeled "Yes " correspond to designs that incorporate xT-STRIDE. The vertical axis shows the probability of a successful attack based on the selected system configuration. The Petri Net simulation results demonstrate that, for both secure and less secure system configurations, the use of xT-STRIDE consistently reduced the probability of a successful cyber attack. These findings confirm that a system designed with xT-STRIDE is more secure than one that does not implement this approach.

Cybersecurity engineers have long recognized the potential of secure system design to prevent most attacks. xT-STRIDE improves this design process using trusted components to achieve targeted CC EAL levels. Simulation results indicate that incorporating such secure systems can lead to substantial improvements in system design. As a result, researchers can have greater confidence in the security of their designs, a confidence that is reflected in the improved outcomes.

7 Conclusion

This study introduces xT-STRIDE, a novel threat modeling method for enhancing the security of UAV systems operating in realistic environments. Security requirements for UAV systems can vary greatly depending on the operational region, influencing factors such as design, cost, and required security functionalities. The effect of incorporating trust levels into the STRIDE model is demonstrated by identifying threats associated with a UAV system using commonly implemented technologies. Analysis across five distinct trust levels reveals considerable variation in threat profiles. For the highest security trust level, approximately 40 threats must be mitigated, while the lowest security trust level requires addressing approximately 120 threats. The disparity in threat numbers directly translates into differences in both economic costs and functional implementations for the UAV system. According to the trust model proposed in this study, any UAV system that lacks classification should disregard the trust levels of its components during the design phase, treating each component equally in terms of security considerations. The key findings of this study are as follows.

Neglecting trust levels during system design leads to inaccurate assessments of security levels. Designers may unintentionally include unnecessary security functionalities in an unclassified UAV system, resulting in high economic costs and reduced functional performance. Additionally, relying on incorrect trust assumptions may cause deficiencies in the established security functionalities, ultimately compromising system security. However, the model ensures that adequate security is provided for every UAV application.
The model enhances security engineers' understanding of threat mitigation by clearly revealing the system's true vulnerability areas. This approach reduces design costs and improves the performance of security components. Furthermore, it does not limit the choice of security frameworks, such as CC, ISO 27001, or others. The solution supports security provisioning during the entire design process.
The study theoretically modeled the UAV from the system level down to the hardware level and demonstrated the model's effectiveness using Petri Net simulation for 5 T-STRIDE.

The xT-STRIDE trust levels do not require a single global trust level for the entire system. Security engineers can effectively implement different xT-STRIDE levels by appropriately partitioning the operating environment. Future research will focus on integrating the developed model into smart city UAV applications, specifically targeting cities with populations over 1,000,000. This will allow for an evaluation of the budgetary implications of applying the xT-STRIDE model to enhance urban security through UAV systems. Additionally, future studies will explore the applicability of xT-STRIDE to new domains, such as robotics and industrial environments.

Author contribution All authors wrote the main manuscript text and prepared figures. All authors reviewed the manuscript.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). Not applicable.

Data availability No datasets were generated or analysed during the current study.

Declarations

Conflicts of interest The authors declare no competing interests.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Mohamed, N., et al.: Unmanned aerial vehicles applications in future smart cities. Technol. Forecast. Soc. Chang. **153**, 119293 (2020). https://doi.org/10.1016/j.techfore.2018.05.004
2. Barmpounakis, E.N., Vlahogianni, E.I., Golias, J.C.: Unmanned Aerial Aircraft Systems for transportation engineering: Current practice and future challenges. International Journal of Transportation Science and Technology **5**(3), 111–122 (2016)
3. Rani, C., et al.: Security of unmanned aerial vehicle systems against cyber-physical attacks. The Journal of Defense Modeling and Simulation **13**(3), 331–342 (2016). https://doi.org/10.1177/1548512915617252
4. Du Du, H., and Heldeweg, M. A. "Responsible design of drones and drone services." (2017). https://doi.org/10.2139/ssrn.3096573
5. RTCA DO 326: A2014 AIRWORTHINESS SECURITY PROCESS SPECIFICATION. (n.d.). Retrieved September 21, 2024, from https://shop.standards.ie/en-ie/standards/
6. EUROCAE, W. (2010). ED202-Airworthiness Security Process Specification.
7. Florence Esselin, Karl Coulon. EBIOS RISK MANAGER : ACCESSIBLE METHODOLOGY TO SECURE DIGITAL TRANSFORMATION. Les Notes du CREOGN, 2021, N° 62. ⟨hal-03813843⟩
8. Voas, J.: Networks of 'things.' NIST Spec. Publ. **800**(183), 800–183 (2016)
9. Standard, I. S. O. (2021). ISO/SAE 21434-Road Vehicles–Cybersecurity engineering.
10. Khan, R., McLaughlin, K., Laverty, D., & Sezer, S. (2017, September). STRIDE-based threat modeling for cyber-physical systems. In 2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe) (pp. 1–6). IEEE.
11. ISO/IEC 15408–1:2022. (2022, November 29). ISO. Retrieved December 1, 2023, from https://www.iso.org/standard/72891.html
12. Jacobsen, R. H. and Marandi, A. (2021). Security threats analysis of the unmanned aerial vehicle system. MILCOM 2021 - 2021 IEEE Military Communications Conference (MILCOM). https://doi.org/10.1109/milcom52596.2021.9652900
13. Gassara, A., Rodriguez, I.B.: Describing correct UAVs cooperation architectures applied on an anti-terrorism scenario. Journal of Information Security and Applications **58**, 102775 (2021). https://doi.org/10.1016/j.jisa.2021.102775
14. Qureshi, K.N., et al.: Trust and priority-based drone assisted routing and mobility and service-oriented solution for the internet of vehicles networks. Journal of Information Security and Applications **59**, 102864 (2021). https://doi.org/10.1016/j.jisa.2021.102864
15. Almulhem, A.: Threat modeling of a multi-UAV system. Transportation Research Part A: policy and practice **142**, 290–295 (2020). https://doi.org/10.1016/j.tra.2020.11.004
16. Singh, K. and Verma, A. K. "Threat modeling for multi-UAV Adhoc networks." TENCON 2017–2017 IEEE Region 10 Conference. IEEE, 2017. https://doi.org/10.1109/TENCON.2017.8228102
17. Shostack, Adam. Threat modeling: Designing for security. John Wiley & Sons, 2014.
18. Gupta, L., Jain, R., Vaszkun, G.: Survey of important issues in UAV communication networks. IEEE communications surveys & tutorials **18**(2), 1123–1152 (2015). https://doi.org/10.1109/COMST.2015.2495297
19. Bouhamed, Omar, et al. "Lightweight ids for uav networks: A periodic deep reinforcement learning-based approach." 2021 IFIP/IEEE International Symposium on Integrated Network Management (IM). IEEE, 2021.
20. Seo, Jong Wan, et al. "Blockchain-Based Secure Firmware Update Using an UAV." Electronics 12.10 (2023): 2189. https://doi.org/10.3390/electronics12102189
21. Mohsan, S.A.H., Othman, N.Q.H., Li, Y., et al.: Unmanned aerial vehicles (UAVs): practical aspects, applications, open challenges, security issues, and future trends. Intel Serv Robotics **16**, 109–137 (2023). https://doi.org/10.1007/s11370-022-00452-4
22. Kim, S., Cho, H., Jung, D.: Circular Formation Guidance of Fixed-Wing UAVs Using Mesh Network. IEEE Access **10**, 115295–115306 (2022). https://doi.org/10.1109/ACCESS.2022.3218673
23. Jahangeer, Mustafa, Z. & Aldabbas, H. Utilizing a Hybrid Cat-Swarm Optimization Algorithm for Seamless Vertical Handoff in Vanet. Wireless Personal Communication,July (2024). https://doi.org/10.1007/s11277-024-11424-5
24. Naskath. Fast multicriteria network selection scheme using hybrid of cat swarm optimization and TOPSIS algorithm for optimal handover in VANET. Trans Emerging Tel Tech. 2023;e4743. https://doi.org/10.1002/ett.4743.
25. "A GAN-based Hybrid Deep Learning Approach for Enhancing Intrusion Detection in IoT Networks", S. Balaji, G. Dhanabalan, C. Umarani and J. Naskath", International Journal of Advanced Computer Science and Applications (IJACSA), 15(6), 2024. – (SCI Indexed)
26. Sonny, A., Yeduri, S.R., Cenkeramaddi, L.R.: Autonomous UAV Path Planning Using Modified PSO for UAV-Assisted Wireless Networks. IEEE Access **11**, 70353–70367 (2023). https://doi.org/10.1109/ACCESS.2023.3293203
27. Naskath, J,Dr. Nithyanantham,M.Pappathi Jancy Rani, "Design Secure Connectivity Protocol to mitigate malicious activity using Game Theory in VANET",PERIODICO di MINERALOGIA,ISSN: 0369–896,Volume 91, No. 5, 2022.
28. "Jawhar, I., et al. ""Communication and networking of UAV-based systems: Classification and associated architectures."" Journal of Network and Computer Applications 84 (2017): 93–108. https://doi.org/10.1016/j.jnca.2017.02.008
29. Tran, Trung Duc, et al. "Methodology for risk management related to cyber-security of Unmanned Aircraft Systems." 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE, 2019. https://doi.org/10.1109/ETFA.2019.8869200
30. Zhang, Yunru, et al. ""A lightweight authentication and key agreement scheme for Internet of Drones."" Computer Communications 154 (2020): 455–464. https://doi.org/10.1016/j.comcom.2020.02.067

31. Ossamah, Almotery. "Blockchain as a solution to drone cybersecurity." 2020 IEEE 6th World Forum on Internet of Things (WF-IoT). IEEE, 2020. https://doi.org/10.1109/WF-IoT48130.2020.9221466
32. Yazdinejad, Abbas, et al. "Enabling drones in the internet of things with decentralized blockchain-based security." IEEE Internet of Things Journal 8.8 (2020): 6406–6415. https://doi.org/10.1109/JIOT.2020.3015382
33. Sharma, Vishal, et al. "Neural-blockchain-based ultrareliable caching for edge-enabled UAV networks." IEEE Transactions on Industrial Informatics 15.10 (2019): 5723–5736. https://doi.org/10.1109/TII.2019.2922039
34. Won, J., Seo, S.-H., Bertino, E.: Certificateless cryptographic protocols for efficient drone-based smart city applications. IEEE Access **5**, 3721–3749 (2017). https://doi.org/10.1109/ACCESS.2017.2684128
35. Ogidan, Ezekiel T., Kamil Dimililer, and Yoney Kirsal-Ever. ""Machine learning for cyber security frameworks: a review."" Drones in Smart-Cities (2020): 27–36. https://doi.org/10.1016/B978-0-12-819972-5.00002-1
36. Wazid, Mohammad, et al. "Design and analysis of secure lightweight remote user authentication and key agreement scheme in internet of drones deployment." IEEE Internet of Things Journal 6.2 (2018): 3572–3584. https://doi.org/10.1109/JIOT.2018.2888821
37. Marchetti, Eda, Tauheed Waheed, and Antonello Calabrò. "Cybersecurity Testing in Drones domain: a Systematic Literature Review." IEEE Access (2024). https://ieeexplore.ieee.org/document/10750190
38. Kumar, Naveen, and Ankit Chaudhary. "Surveying cybersecurity vulnerabilities and countermeasures for enhancing UAV security." Computer Networks 252 (2024): 110695. https://www.sciencedirect.com/science/article/abs/pii/S1389128624005279
39. Miao, Shangting, Quan Pan, and Dongxiao Zheng. "Unmanned aerial vehicle intrusion detection: Deep-meta-heuristic system." Vehicular Communications 46 (2024): 100726. https://www.sciencedirect.com/science/article/abs/pii/S2214209624000019
40. Liu, Guangshuai, et al. "Design of a multi-component system-based fixed-wing unmanned aerial vehicle maintenance policy and its case study." Computers & Industrial Engineering (2024): 110701. https://www.sciencedirect.com/science/article/abs/pii/S0360835224008234
41. Zhao, Dongmei, et al. "Security situation assessment in UAV swarm networks using TransReSE: A Transformer-ResNeXt-SE based approach." Vehicular Communications 50 (2024): 100842. https://www.sciencedirect.com/science/article/abs/pii/S2214209624001177
42. Chandran, Indu, and Kizheppatt Vipin. "A PUF secured lightweight mutual authentication protocol for multi-UAV networks." Computer Networks 253 (2024): 110717. https://www.sciencedirect.com/science/article/abs/pii/S1389128624005498
43. Yang, Hanlin, Yajun Guo, and Yimin Guo. "Fault-tolerant security-efficiency combined authentication scheme for manned-unmanned teaming." Computers & Security 146 (2024): 104052. https://www.sciencedirect.com/science/article/abs/pii/S0167404824003572
44. Zhou, Li, et al. "A Comprehensive Survey of Artificial Intelligence Applications in UAV-Enabled Wireless Networks." Digital Communications and Networks (2024). https://www.sciencedirect.com/science/article/pii/S2352864824001536
45. Li, Zhihao, et al. "A secure and efficient UAV network defense strategy: Convergence of blockchain and deep learning." Computer Standards & Interfaces 90 (2024): 103844. https://www.sciencedirect.com/science/article/abs/pii/S0920548924000138
46. Akram, Muhammad Arslan, et al. "Blockchain-based privacy-preserving authentication protocol for UAV networks." Computer Networks 224 (2023): 109638. https://www.sciencedirect.com/science/article/abs/pii/S138912862300083X
47. Zhao, Xu, et al. "SAC-based UAV mobile edge computing for energy minimization and secure data transmission." Ad Hoc Networks 157 (2024): 103435. https://www.sciencedirect.com/science/article/abs/pii/S1570870524000465
48. OWASP, S. Software assurance maturity model version 2. (2022).
49. Microsoft. "Simplified Implementation of the Microsoft SDL." (2010).
50. Thuraisingham, B. and Hamlen, K. W. . "Challenges and future directions of software technology: Secure software development." 2010 IEEE 34th Annual Computer Software and Applications Conference. IEEE, 2010. https://doi.org/10.1109/COMPSAC.2010.88
51. Kara, M.: Review on common criteria as a secure software development model. International Journal of Computer Science & Information Technology **4**(2), 83 (2012). https://doi.org/10.5121/ijcsit.2012.4207
52. Solutions, C.-W.D. Curtiss-Wright Defense Solutions Data Transport System 1-Slot Hardware Encryption Layer version 5.1 (FDEEEcPP20E/FDEAAcPP20E) Security Target. 2020; Available from: https://commoncriteriaportal.org/files/epfiles/st_vid11096-st.pdf.
53. EADS-CASA. EADS Air Segment System Protection Profile. 2011; Available from: https://www.commoncriteriaportal.org/files/ppfiles/2010-35-INF-627.pdf.
54. Board, C.C. Common Evaluation Methodology CEM. 2022 Rev1. Retrieved March 1, 2023. Available from: https://www.commoncriteriaportal.org/files/ccfiles/CEM2022R1.pdf
55. Ozmen, Muslum Ozgur, and Attila A. Yavuz. "Dronecrypt-an efficient cryptographic framework for small aerial drones." MILCOM 2018–2018 IEEE Military Communications Conference (MILCOM). IEEE, 2018. https://doi.org/10.1109/MILCOM.2018.8599784
56. Feng, Zhiwei, et al. ""Efficient drone hijacking detection using two-step GA-XGBoost."" Journal of Systems Architecture 103 (2020): 101694. https://doi.org/10.1016/j.sysarc.2019.101694
57. Abdi, Fardin, et al. "Guaranteed physical security with restart-based design for cyber-physical systems." 2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS). IEEE, 2018. https://doi.org/10.1109/ICCPS.2018.00010
58. Zhu, Haibei, et al. "Operator strategy model development in UAV hacking detection." IEEE Transactions on Human-Machine Systems 49.6 (2019): 540–549. https://doi.org/10.1109/THMS.2018.2888578
59. Abdi, Fardin, et al. "Preserving physical safety under cyber attacks." IEEE Internet of Things Journal 6.4 (2018): 6285–6300. https://doi.org/10.1109/JIOT.2018.2889866
60. Cheon, Jung Hee, et al. "Toward a secure drone system: Flying with real-time homomorphic authenticated encryption." IEEE access 6 (2018): 24325–24339. https://doi.org/10.1109/ACCESS.2018.2819189
61. Ch, Rupa, et al. ""Security and privacy of UAV data using blockchain technology."" Journal of Information security and Applications 55 (2020): 102670. https://doi.org/10.1016/j.jisa.2020.102670
62. Senol, S., Leblebicioglu, K., Schmidt, E.G.: INtERCEDE: An algorithmic approach to networked control system design. J. Netw. Comput. Appl. **34**(4), 1326–1341 (2011). https://doi.org/10.1016/j.jnca.2011.03.018
63. Altawy, R., Youssef, A.M.: Security, privacy, and safety aspects of civilian drones: A survey. ACM Transactions on Cyber-Physical Systems **1**(2), 1–25 (2016). https://doi.org/10.1145/3001836
64. Yaacoub, J.P., et al.: Security analysis of drones systems: Attacks, limitations, and recommendations. Internet of Things **11**, 100218 (2020). https://doi.org/10.1016/j.iot.2020.100218
65. Gupta, Maanak, et al. "Security and privacy in smart farming: Challenges and opportunities." IEEE access 8 (2020): 34564–34584. https://doi.org/10.1109/ACCESS.2020.2975142

66. Sanjab, A., Saad, W., Başar, T.: A game of drones: Cyber-physical security of time-critical UAV applications with cumulative prospect theory perceptions and valuations. IEEE Trans. Commun. **68**(11), 6990–7006 (2020). https://doi.org/10.1109/TCOMM.2020.3010289
67. Sedjelmaci, Hichem, et al. "An efficient cyber defense framework for UAV-Edge computing network." Ad Hoc Networks 94 (2019): 101970. https://doi.org/10.1016/j.adhoc.2019.101970
68. Sharma, Vishal, et al. "Behavior and vulnerability assessment of drones-enabled industrial internet of things (iiot)." IEEE Access 6 (2018): 43368–43383. https://doi.org/10.1109/ACCESS.2018.2856368
69. Yaacoub, Jean-Paul A., et al. ""Cyber-physical systems security: Limitations, issues and future trends."" Microprocessors and microsystems 77 (2020): 103201. https://doi.org/10.1016/j.micpro.2020.103201
70. Tsao, K.Y., Girdler, T., Vassilakis, V.G.: A survey of cyber security threats and solutions for UAV communications and flying ad-hoc networks. Ad Hoc Netw. **133**, 102894 (2022)
71. Albalawi, M., & Song, H. (2019, April). Data security and privacy issues in swarms of drones. In 2019 Integrated Communications, Navigation and Surveillance Conference (ICNS) (pp. 1–11). IEEE.
72. The CORAS Method. (2023, March 16). https://coras.sourceforge.net/
73. Mehari. (2023, March 16). https://www.enisa.europa.eu/topics/risk-management/current-risk/risk-management-inventory/rm-ra-methods/m_mehari.html
74. Monteuuis, JP., et al. "Sara: Security automotive risk analysis method." Proceedings of the 4th ACM Workshop on Cyber-Physical System Security. 2018. https://doi.org/10.1145/3198458.3198465
75. NIST. (2018). Systematic Risk Analysis Method (SRAM). Retrieved from https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-40r2.pdf
76. Security Risk Management Institute. (2017). Risk and Threat Analysis Framework (RTAF). Retrieved from https://srmiportal.org/rta/rta-overview/
77. ISO. (2020). Risk Management Framework (RMF). Retrieved from https://www.iso.org/standard/70102.html
78. Blank, R., and P. Gallagher. "Guide for conducting risk assessments NIST special publication 800–30 Revision 1 JOINT TASK FORCE TRANSFORMATION INITIATIVE." (2012). https://doi.org/10.6028/NIST.SP.800-30r1
79. ISO/IEC 27005:2022. (2022, November 29). ISO. Retrieved December 1, 2023, from https://www.iso.org/standard/80585.html
80. DREAD Researchgate.net. [cited 2021 Jun 6]. Available from: https://www.researchgate.net/publication/332591691_Threat_Modeling_of_Internet_of_Things_Health_Devices
81. Pape, N., & Mansour, C. (2024). PASTA Threat Modeling for Vehicular Networks Security. 2024 7th International Conference on Information and Computer Technologies (ICICT), 474–478. https://doi.org/10.1109/ICICT62343.2024.00083
82. Yadav, T., & Rao, A. M. (2015). Technical Aspects of Cyber Kill Chain. In J. H. Abawajy, S. Mukherjea, S. M. Thampi, & A. Ruiz-Martínez (Eds.), Security in Computing and Communications (pp. 438–452). Springer International Publishing. https://doi.org/10.1007/978-3-319-22915-7_40
83. Alberts, C. J., Behrens, S. G., Pethia, R. D., & Wilson, W. R. (1999). Operationally critical threat, asset, and vulnerability evaluation (OCTAVE) framework, Version 1.0.
84. Ahmed, M., Panda, S., Xenakis, C., & Panaousis, E. (2022, August). MITRE ATT&CK-driven cyber risk assessment. In Proceedings of the 17th International Conference on Availability, Reliability and Security (pp. 1–10).
85. Khan, Rafiullah, et al. "STRIDE-based threat modeling for cyber-physical systems." 2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe). IEEE, 2017. https://doi.org/10.1109/ISGTEurope.2017.8260283
86. Rouland, Quentin, Brahim Hamid, and Jason Jaskolka. ""Specification, detection, and treatment of STRIDE threats for software components: Modeling, formal methods, and tool support."" Journal of Systems Architecture 117 (2021): 102073. https://doi.org/10.1016/j.sysarc.2021.102073
87. Sattar, Danish, et al. "A stride threat model for 5g core slicing." 2021 IEEE 4th 5G World Forum (5GWF). IEEE, 2021. https://doi.org/10.1109/5GWF52925.2021.00050
88. Hussain, Shafiq, Harry Erwin, and Peter Dunne. "Threat modeling using formal methods: A new approach to develop secure web applications." 2011 7th International Conference on Emerging Technologies. IEEE, 2011. https://doi.org/10.1109/ICET.2011.6048492
89. Shakhatreh, H., et al.: Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges. Ieee Access **7**, 48572–48634 (2019). https://doi.org/10.1109/access.2019.2909530
90. Tvaronavičienė, M., et. al. "Cyber security management model for critical infrastructure protection." Proceedings of the Selected papers of the International Scientific Conference "Contemporary Issues in Business, Management and Economics Engineering. 2021. https://doi.org/10.3846/cibmee.2021.611
91. Secure by Design. (n.d.). Retrieved September 21, 2024, from https://ieeexplore.ieee.org/document/10280236
92. The National Institute of Standards and Technology (NIST) The National Institute of Standards and Technology (NIST). (2023, March). [National Vulnerability Database]. Retrieved March 17, 2023, from https://nvd.nist.gov/general/nvd-dashboard