This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 861111

Ref. Ares(2021)3579934 - 31/05/2021

Drones4Safety

Research & Innovation Action (RIA)

Inspection Drones for Ensuring Safety in Transport Infrastructures

Multi-drone system threat analysis and specification of the security system design

D5.2

Due date of deliverable: 31.05.2021 (M12)

Start date of project: June 1st, 2020

Type: Deliverable
WP number: WP5

Responsible institution: Aarhus University (AU)

Editor and editor's address: Rune Hylsberg Jacobsen (rhj@ece.au.dk)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 861111

Version 0.5
Release Date: May 31, 2021

Project funded by the European Commission within the Horizon 2020 Programme		
Dissemination Level		
PU	Public	✓
CO	Confidential, only for members of the consortium (including the Commission Services)	

Authors/contributors

The following members of the project have contributed to the content and writing of this deliverable:

- Rune Hylsberg Jacobsen (AU, editor)
- Lea Matlekovic, University of Southern Denmark (SDU IMADA)
- Sam Münchow, Automotive & Rail Innovation Center GmbH (ARIC)

Change Log

Rev.	Date	Who	Site	Change
0.1	01/06/2020	Annika Lindberg	SDU	Created initial version.
0.4	31/05/2021	Rune H. Jacobsen	AU	First draft on version.

1 Executive Summary

There is and increasing uptake of autonomous drones for inspection in the industry. The Drones4Safety (D4S) project is investigating the use of a swarm of autonomous drones connected to the Internet via a ground station or through a mobile network for inspection applications. The security of these type of systems is of utterly importance. On the one side, a malicious user intervening or taking control of the inspection drones can lead a disastrous crashing of the drones into ground infrastructures or even hitting humans. On the other hand, an adversary may maliciously overtake the drone system of parts of it to capture valuable assets.

This deliverable provides an analysis of threats and vulnerabilities of the D4S multi-drone system in the context of the defined inspection use cases in the project. We base the methodology for the analysis in the STRIDE model. This model ensures that the analysis addresses a broad view of threats as it classifies threats into spoofing, tampering, repudiation, information disclosure, denial-of-service, and elevation of privilege attacks. The identified threats are subsequently used as input in a threat analysis to prioritize them according to severity, i.e., the product of likelihood and impact.

We conclude by identifying the most important threats to counter in the multi-drone system and establish a set of recommendations for future versions of the D4S multi-drone system.

Contents

Acronyms

Acronym	Description
APP	Application part (software)
C2	Command and Control
COM	Communication interface
COTS	Common Of The Shelf
CS	Cloud Services
CSIRT	Computer Security Incident Response Team
D4S	Drones4Safety
DoS	Denial-of-Service
DDoS	Distributed Denial-of-Service
DFD	Data flow diagram
DSP	Digital Signal Processor
ENISA	European Union Agency for Cybersecurity
FDR	Flight Data Recorder
GCS	Ground Control Station
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
ICT	Information Communication Technology
IMU	Inertial Measurement Unit
INF	Computing Infrastructure
IP	Internet Protocol
MANET	Mobile Ad Hoc Network
MECH	Mechanical part
OBC	On-board Computer
OS	Operating System (Operating System part)
POW	Power distribution part
RF	Radio Frequency
ROS	Robotic Operating System
SEN	Sensor part
SSL	Secure Socket Layer
TCP	Transmission Control Protocol
TLS	Transport Layer Security

UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicles
UDP	User Datagram Protocol
UML	Unified Modeling Language
USB	Universal Serial Bus
WSN	Wireless Sensor Network

2 Introduction

The use of Unmanned Aerial Vehicles (UAVs) in the airspace systems has been characterized as the next great step forward in the evolution of civil aviation. Although use of unmanned aircraft systems (UAS) in public service operations is thriving, civil use of UAS remains limited considering its huge potential societal impacts. During the past century breakthrough in research and innovation in autonomous robotics and UAV systems, or know with the more popular term: drone systems, has led to and increased deployment of drones in European airspace. Drones typically perform tasks such as surveillance, crop monitoring, imaging, surveying, search and rescue, infrastructure inspections and military use. Many of these drones execute their duties entirely autonomously without direct intervention of human operators. Due to the nature of the responsibilities of drones, their security is of utmost importance.

From a security point of view, the threat landscape is a collection of threats in a particular domain or context, with information on identified vulnerable assets, threats, risks, threat actors and observed trends [15]. Threat landscapes can be broad, including the entire range of cyber threats, or targeted at a particular sector, such as the financial sector, critical infrastructure, smart homes, or similar. Security threats to drones are typically targeted at the UAV system-level, which includes everything employed to allow the drone to function. This may include the hardware and software running on the drone, the ground control system piloting the drone and the connection between the two.

In 2009, the media reported that a US drone had been hacked by Iraqi insurgents [4]. Insurgents were able to intercepted live video feeds from the drones being relayed back to a US controller and revealing potential targets by using cheap internet software. The problem was only disclosed after the US military found many hours' worth of video recordings on militant laptops. Other known attacks to Unmanned Aerial Systems (UASs) includes computer virus inspection of the mission command center in US air force base controlling military drones [5].

Researchers at Check Point Software Technologies Ltd. and DJI, recently shared details of a potential vulnerability that could have affected DJI's infrastructure, if exploited [6]. The researchers outlined a process in which an attacker could gain access to user accounts via a vulnerability discovered in the user identification process within a DJI cloud service (DJI forum).

The aim of this deliverable is to provide a security threat and vulnerability analysis of the multi-drone system designed and developed in WP5 of the Drones4Safety (D4S) project. The result of the analysis provided herein supports manufacturer of drone and robotics components to provide an understanding of the components needed in an autonomous multi-drone system and to allow them to identify and address security concerns within their systems in a prioritized manner.

2.1 Security objectives

The long-term commercial viability of the D4S inspection solution will heavily depend on the level of trust that can be provided to the end-users. As a first step, the D4S project needs to create awareness on system threats and point to possible mitigation and protections mechanisms. To this end, the design of the multi-drone system shall adhere to the following high-level aims:

- To ensure that information generated by or relating to an inspection drone is adequately protected against misuse or misappropriation;
- To ensure that the resources and services provided by the inspection services is adequately protected against misuse or misappropriation;
- To ensure that the security features are adequately compliant with standardized to ensure world-wide interoperability between different system components;

- To ensure that the security features compatible with world-wide availability, i.e., there should be at least one ciphering algorithm that can be exported on a world-wide basis in compliance with the current international export regulations;
- To ensure that the implementation of security features and mechanisms can be extended and enhanced as required by new threats and services;
- To ensure that intellectual property of the system is protected against industrial espionage.

The drone platform, the platform of the GCS as well as the cloud-computing platform will be based on open hardware, Common Of The Shelf (COTS) components, and open software components and distributions such as the Linux Operating System (OS). Therefore, the above high-level security objectives need to be enforced and validated towards a third party e.g., a subsystem/component vendor or a service provider.

The above objectives together can be met by provision of methods to achieve the following security general objectives [10]:

- **Confidentiality:**
  The avoidance of the disclosure of information without the permission of its owner.
- **Integrity:**
  The property that data has not been altered or destroyed in an unauthorized manner.
- **Authentication:**
  The property of ensuring that a communicating entity is the one that it claims to be.
- **Authorization:**
  The property of giving or preventing access to a system resource.
- **Availability:**
  The property of being accessible and usable upon demand by an authorized entity.
- **Non-repudiation:**
  The property by which one of the entities or parties in a communication cannot deny having participated in the whole or part of the communication.

Many of these security goals can be achieved by the propose configuration and use of already implemented security features of third party products. As an example, Table 1 provides and overview of the essential security support available in Linux.

Table 1: Basic security features in Linux.

Security features in Linux
• User authentication and authorization • Encryption, • Support for trusted computing platform/secure element • Secure network layer • Secure application layer channel between elements of the multi-drone system • Firewall and packet filtering • Wireless security support

2.2 Related work

The European Union Agency for Cybersecurity (ENISA) is the Union's agency dedicated to achieving a high common level of cybersecurity across Europe. Among others, the agency contributes to EU cyber policy, by identifying and evaluating the top cyber threats on a regular basis with the aim of enhances the trustworthiness of ICT products, services and processes [15]. To this end, ENISA has addressed thematic threat landscapes related to internet infrastructure, smart grids, and 5G networks of relevance to the D4S project.

In [1] the authors provides an overview of autonomous UAS architecture and analyzes security threats to the system. The work focuses on a vulnerability assessment of the non-autonomous and long range autonomous UAS. The paper states that the placement of GCS open a new range of attacks that are not typical in most UAS deployments [1].

In [8], the authors discuss various security threats to a UAV system. Threats are analysed and a cyber-security threat model showing possible attack paths is proposed. In this, work the security triad, with its three facets: Confidentiality, Integrity and Availability, was used as the overarching security model. Furthermore, the paper focused on communication model. It is argued that from a security and threat analysis perspective, it is necessary to understand that a typical UAV network is not similar to the traditional computer network.

Some researchers have compared the security model for a UAS to wireless sensor networks (WSNs) [12] and mobile ad-hoc networks (MANETs) [20]. Although these network shows close similarity to WSNs, as both of them use wireless communication protocols [13], there are other characteristics in which they differ. For instance, power requirements, amount of information being carried by channels, and the number of nodes in a WSN are much lower than in a networks of UAVs. Moreover, the coverage area for a UAS can be about thousand times bigger than that of a WSN. Moreover, all nodes of a WSN usually transmit their sensor data to one central node which communicates with external systems, in a network of UAVs, each of the UAVs needs continuously to communicate with the GCS.

3 System model

3.1 Security model

A security model is a framework for specifying and enforcing security policies for a system. Aside from enforcement of security, the security model also provides a reference for a systematic approach to analyzing threats and system vulnerabilities. The security model enables us to address specific security concerns and mitigations from different perspectives such as the different security objectives defined (cf. Sec. 2.1).

Security needs to be addressed from a system point of view as security cannot be stronger as the weakest link of the system. In other words, security crosses interfaces between components and across stakeholder organizations. To make the analysis of security more manageable it is often advantageous to look at the taxonomy of a system from a functional point of view. The system taxonomy, which is a scheme for a hierarchical classification to organize the system into different groups/types of constituents.

3.2 Taxonomy

Our system-of-interest i.e., the multi-drone system is a subsystem of an unmanned aircraft system (UAS), which include additionally a Ground Control Station (GCS) and a system of communications with an Unmanned Aerial Vehicle (UAV), i.e., a drone. In the following, we analyze the system-of-interest from its structural and behavioral (operations) point of view with the aim to provide a taxonomy for the system. The taxonomy is used to structure the subsequent threat analysis by providing a breakdown structure of the system into subsystem/component levels.

Deliverable D2.4 provided an analysis of different inspection sites of interest for the D4S project to find specific use-cases for bridge and railway inspection suitable to test and validate the D4S platform including the multi-drone swarm system [11]. A more elaborate specification of the multi-drone system can be found in [14].

During inspection missions, the multi-drone system will be deployed as a set of autonomous, collaborative drones. Each drone will be assigned a sequence of tasks e.g., inspecting an element of the target object, charging, etc. Each drone will be monitored by using a GCS. Communication between drones is needed to ensure coordination. The inspection mission is supported by a set of services deployed in the backend cloud infrastructure. Such digital services includes mission supervision from a Control Station and storage of inspection images (payload data).

To analyze the security threats of the multi-drone system in the context of the two types of inspection use cases addressed in the D4S project [11], we provide a breakdown of the system into its main subsystems.

Figure 1 shows the overarching system concept of the D4S project.

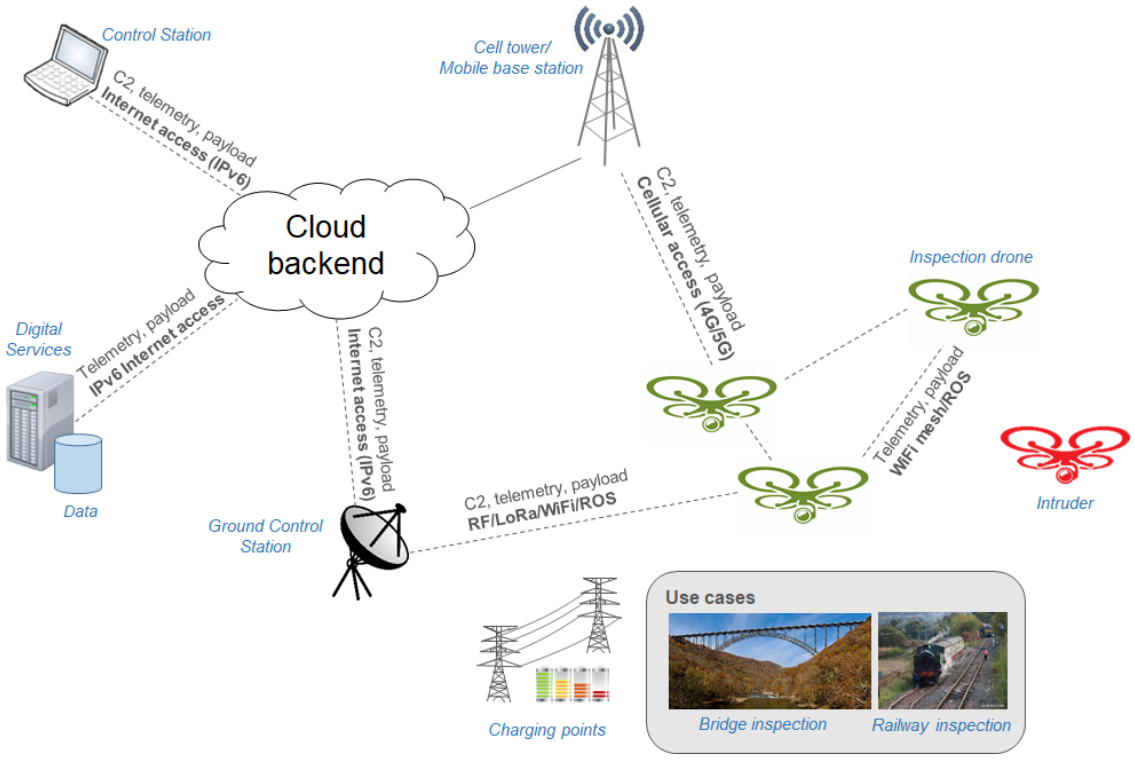

Figure 1: System overview for the multi-drone system in operation.

The multi-drone system forms a wireless mesh network (WiFi mesh) to support collaboration between drones. The drone mesh network also acts to extend the coverage that can be served by a GCS. Each drone is connected to a GCS for supervisory control through a command and control channel (C2). For this purpose long range LoRa wireless communication or by using proprietary Radio Frequency (RF) communication. This network setup ensure a level of redundancy because each drone is equipped with at least two different types of radio interfaces. Furthermore, it allow telemetry and payload data to be routed over multiple hops in the network.

The UAS contains a set of digital support services such as mission planning, data storage etc. provided through a cloud backend in [14]. Although dedicated network access is possible, it is anticipated that access to the cloud backend typically will obtained by using public Internet services to reduce overall system cost. In case

there is 4G or 5G mobile network coverage at the inspection area, drones may optionally be equipped with data modems to connect to cloud services of the UAS.

Figure 2 shows the taxonomy of our UAS. On the top-tier (Tier 1) we are addressing the UAS generally. As we progress the tier levels, the breakdown structure becomes more specific. On Tier 2, we have the multi-drone system i.e., the UAVs, the Ground Control System (GCS) and the Cloud Services (CS). All these elements are needed to carry out the inspection use cases of the D4S project. Tier 3 contains the components composing the subsystems on Tier 2. These include mechanical parts, sensors, communication and power elements as well as operating system and software applications.

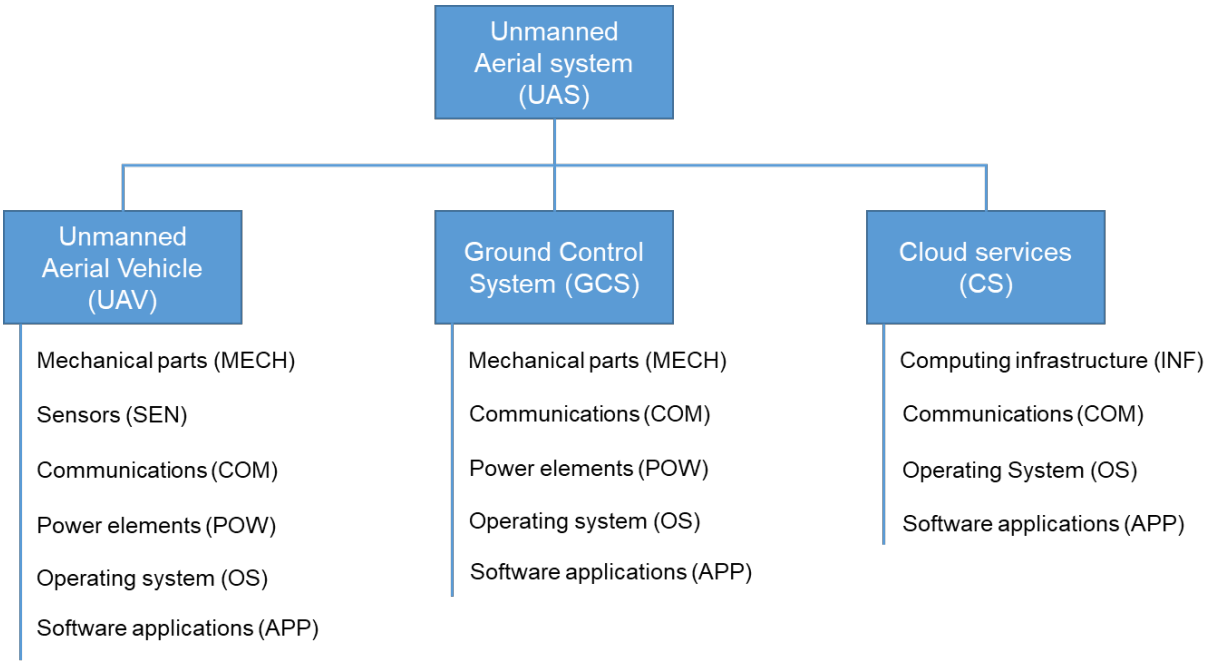

Figure 2: Taxonomy of the UAS.

3.2.1 The Unmanned Aerial Vehicle (UAV) subsystem

In D4S, we consider rotary-wing drones (UAVs) for inspection due to the high maneuverability. An UAV is composed of mechanical (MECH) parts such as the body frame, electrical motors, propellers and antennas. The UAV is further equipped with sensors (SENs) to support the inspections and well as the drone navigation e.g., IMUs, a GNSS receiver, magnetometers, cameras, and possible also LIDARs. An operating system (OS) including flight-controllers and on-board computing (OBC) platforms support the running of software applications (APPs). Furthermore, the UAV subsystem embeds a number of communication interfaces (COMs) accessible through antenna systems and through electronic ports such as USB. A power distribution system (POW) ensures power to be delivered to relevant mechanical parts as well as the computing systems. Moreover, the POW offers a way to harvest energy from a charging point.

It is important to note that UAVs have limited memory and computation capability as compared to the GCS and in particular CS [21].

3.2.2 Ground Control Station (GCS) subsystem

Multiple UAVs may connect to a single GCS. Multiple GCS can be deployed to serve a single or multiple UAVs. The GCS takes flight commands from the pilot and transmits them wirelessly to the flight controller and the OBC of the UAV. UAVs communicate with the ground control station through C2 links of the COM.

Although always wireless, different mediums such as radio frequency (RF), cellular network, satellite, and WiFi are used to support a variety of communication interfaces (COMs). Furthermore, the GCS is an IP node that allows multi-drone system to connect to the Internet. Like the UAV, the GCS is composed of mechanical (MECH) parts such as a ruggedized computer container, antennas, and possible also a display. A computing platform with operating system (OS) supports the running of application (APPs) on the GCS. The GCS will typically be battery powered (POW) to ensure that it can be carried in the field.

3.2.3 Cloud Services (CS) subsystem

A backend cloud infrastructure is used for payload storage and processing and the display of mission data. The cloud infrastructure stores sensitive information about the customer, their missions, and the payload results. For this reason, the security and privacy of the cloud is an important aspect of the UAS threat analysis. Access to cloud services is provided over an IP infrastructure (COM). This allows us to deploy standardized cybersecurity protection from the Internet society such as firewalls, intrusion detection and prevention systems as well as protocols for protecting application data e.g., TLS/SSL. A key aspect of the cloud services is to provide a scalable storage and computation infrastructure (INF). The continuous collection of high-resolution inspection images and possible videos will require large and resilient data storage capacity. Furthermore, algorithms in data analysis and machine learning calls for upscaling of the computing infrastructure.

3.3 Attack surface

The attack surface of the multi-drone system may be considered as the combination of the different points for cyberattacks, where a malicious user (i.e., the "attacker") can try to enter data to or extract data from the system. Keeping the attack surface as small as possible is a basic security measure.

The attack surface of the multi-drone system can be divided into attack made possible by

- Gaining (capturing) physical access to one or more drones or ground control stations (GCSs)
- Intervening, intercepting or otherwise exploiting a communication interface
- Manipulation with the software installed in a drone, in the cloud software or in the GCS.

Concerning communication, one may further differentiate between data traffic that is used for control and data resulting from the mission operations. Examples of control data includes command & control information for operating the drones, control data to maintain connectivity such as routing information or data to maintain security such as key exchange.

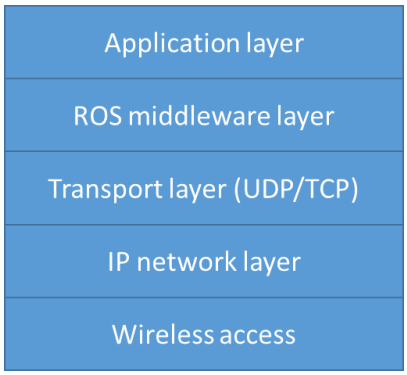

Figure 3: D4S protocol stack - simplified view.

It is important to note that cyberattacks on communication interfaces can take place at different layers in the protocol stack (Figure 3). Obviously, a wireless interface can be intercepted from the knowledge of the radio frequencies in use and with a proper selection of radio equipment and antennas. As the multi-drone system connects to the Internet, among other to get access to Cloud Services (CS), the system is exposed to communication over IP networks and the vulnerabilities associate with this. There is a good knowledge base published concerning threats related to TCP/IP communication (see e.g., [23]) and vulnerabilities on the Internet are continuously monitored by organization such as CSIRT [24]. The increasing use of ROS for autonomous robotic systems in the industry exposes for a new set of communication interfaces [25].

Software security is a long-standing challenge to digital industries. A software bug may lead the software to act differently from the intended behaviour as defined in the software specification. This may result in a vulnerability that can be exploited by a malicious user [26]. Furthermore, "backdoors" probably exists in software [27]. This vulnerability is further stressed by the large amount of open source software used in the multi-drone system. This is a common practice throughout digital industries and natural consequence of the complexity of modern computer systems. Executing unknown code has several security implications among others it involves the question whether the executable software has some un-documented, unwanted, or hidden functionality [27].

4 Threat analysis framework

This section introduces the methodology used to analyze threats and vulnerabilities of the D4S UAS. It introduces the STRIDE model, which is an emerging technique for threat analysis of digital information systems [2]. Furthermore, we introduce the data flow diagram, which is a useful method for understanding the behavior of a system from the point of view of data flowing between subsystems and components of the system. Towards the end of the section, we describe our method to evaluate risk, which is essential for providing a prioritization of security threats.

4.1 Stride model

To define the requirements and derive potential solutions to secure the drone inspections, the WP5 team have applied the STRIDE methodology to collected input from the use case owners and compiled the results. STRIDE was developed by Microsoft and used for analysis of computer security threats at first [2][3]. The STRIDE methodology can also be used for other systems. The name comes from the initials of the different threat categories that the model covers. Table 2gives an overview on the different threats and the according security properties.

Table 2: Explanation of the STRIDE threat methodology. Adapted from [2].

Threat category	Security objective	Description
Spoofing	Authentication	Pretending to be something or someone other than yourself. This includes the illegally accessing and then using another user's authentication information, such as username and password.
Tampering	Integrity	Data tampering involves the malicious modification of data or other parts of the system.
Repudiation	Non-Repudiation	Claiming that you did not do something, or were not responsible. Repudiation threats are associated with users who deny performing an action without other parties having any way to prove otherwise. Nonrepudiation refers to the ability of a system to counter repudiation threats.
Information disclosure	Confidentiality	Providing information to someone not authorized to see it: data leak or privacy breach
Denial of Service (DoS)	Availability	Denial of service (DoS) attacks deny service to valid users by e.g., absorbing resources needed to provide service.
Elevation of privilege	Authorization	Here an unprivileged user gains privileged access and thereby has sufficient access to compromise, destroy or get access to do something they are not authorized to.

To analyze a D4S multi-drone system, a system overview is created that contains the relevant processes and communication links. The applicable STRIDE threats are then identified on system level for every entity. Afterwards a more detailed look is taken at the involved processes and other entities.

1. First, we identify the assets of an entity of the system. Those assets should be directly coupled to the entity.
2. The next step is to define threats that can harm the defined assets. This is done on a high-level using the STRIDE categories. Tampering of data can for example be a threat, if data integrity is an asset. How data can be tampered is not part of this analysis.
3. A risk assessment is performed for each of the defined threats to identify the potential impact and therefore be able to identify the most crucial threats.

The results of this STRIDE analysis will later be used to identify the security architecture and mitigation technologies in D4S. In the following, the threats and risk rating for the most important processes and communication links of the use cases will be listed.

Figure 4 illustrates the workflow conducted to elicit and prioritize threats for the multi-drone system.

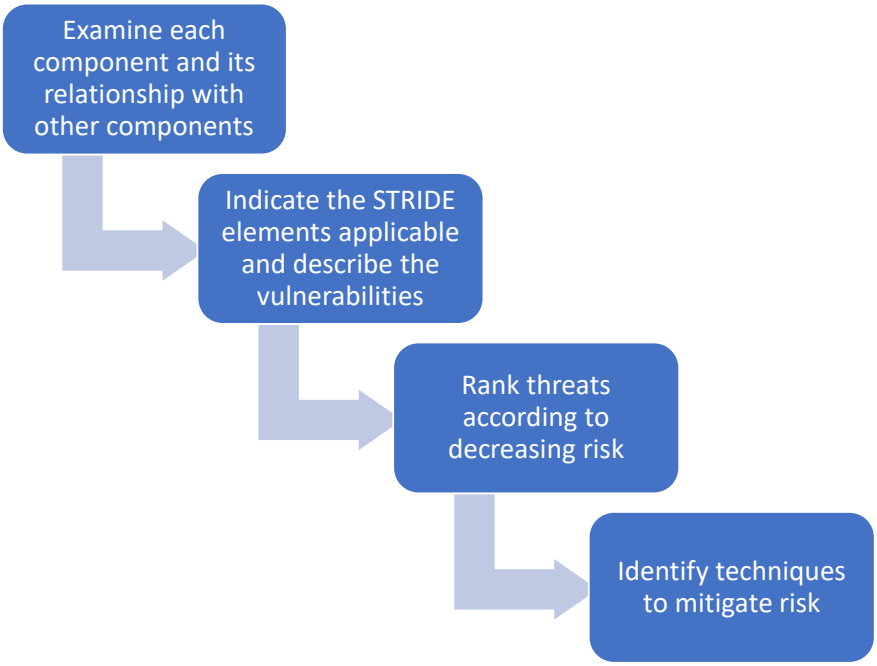

Figure 4: Workflow for the threat analysis.

To follow STRIDE, we decompose our system into relevant components, analyze each component for susceptibility to the threats, and address these threats individual with the aim to propose way to mitigate these. Threats are ranked according to a risk assessment (see Section 4.3). The risk assessment yields the risk-level as a product of the assessed scoring of the likelihood and impact of the risk. The process is repeated until we are comfortable with any remaining threats. Eventually, we manage to break our system down into components and mitigate all the threats to each component. However, this is be itself not sufficient to prove that the system is secure since we have not yet shown that the interactions between components and subsystems are individually immune to a spoofing threat are not susceptible to a spoofing threat when they are combined into a system. In fact, threats may materialize only when subsystems are joined to create larger systems. In most of

those cases, the very act of combining subsystems into larger systems involves violating the original assumptions the subsystem made.

Table 2 also provides a linking of the STRIDE elements with the security objectives from Section 2.1.

4.2 Data flow datagrams

Data flow diagrams (DFDs) may be used to graphically represent the interaction of subsystem components in integrated system. The main objectives to use data flow diagram representations in the threat analysis is to ensure threat that arises in the interplay between different subsystems are properly addressed.

DFDs use a standard set of symbols consisting of four elements: data flows, data stores, processes, and interactors, and for threat modelling, we add one more—trust boundaries. Table 3 lists the symbols. Data flows represent data in motion over network connections etc. Data stores represent files, databases, registry keys, and so no. Processes are computations, algorithms or programs (software) run by the computer.

Table 3: List of items and symbols of a data flow diagram.

Item	Symbol
Data flow	One way arrow
Data store	Two parallel horizontal lines
Process	Circle
Multi-process	Two concentric circles
Interactors	Rectangle
Trust boundary	Dotted line

4.3 Risk evaluation

Threats are analyzed with respect to their likelihood of occurrence, their possible impact on individual users and system and the global risk they represent. Table 4 shows the Risk Evaluation Grid, which is used for the threat analysis of various threats and has been defined as a standard grid in the ETSI threat assessment methodology [10]. The evaluation is conducted according to three criteria: *Likelihood*, *Impact* and *Risk*.

This grid helped us in our detailed analysis. The analysis helps us in determining the likelihood, impact and risk of each of the possible threats and gives us an insight in the overall possible damage to the system due to a particular threat.

The Likelihood score evaluates the possibility of attacks being initiated. It is *Unlikely* (1) if a potential attacker has much less information and needs to resolve several technical difficulties, or if there is a low motivation. It is *Possible* (2) if there are lesser or no technical problems, or if there are several reasons for someone to launch an attack and it is *Likely* (3) if there is a high motivation and no problems.

The Impact score signifies the resulting state of the system after an attack. It is *Low* (1) if the attack creates only low-level problems, and the problems created are usually reversible and repairable. It is *Medium* (2) if the attack is directed to loss of service of a single user for a considerable amount of time or limited scope outage for a multi-user system. The Impact is High (3) if the attack directed to a single user causes a loss of service for a long period of time or longer periods of outages with many users being affected and possible law

violations or financial losses. The Likelihood and Impact vary from one to three as shown in Table 4. the Risk is calculated as the product of the Impact and Likelihood values For a given threat.

Table 4: Risk evaluation grid.

		Rationale		
Criteria	**Cases**	**Difficulty**	**Motivation**	**Ranks**
Likelihood	Unlikely	Strong	Low	1
	Possible	Solvable	Reasonable	2
	Likely	None	High	3
		User	***System***	
Impact	Low	Annoyance	Very limited outage	1
	Medium	Loss of service	Limited outage	2
	High	Long time loss of service	Long term outage	3
Risk	Minor	No need for countermeasures		1,2
	Major	Threat need to be handled		3,4
	Critical	High priority		6,9

5 Threat analysis

In this section, we will discuss threats in detail using the STRIDE Threat Model. Our analyses is organized according to the STRIDE threat categories. It is a synthesis of a data collection process in WP5, where work package members have given their input to set of foreseeable security threats.

5.1 Spoofing

The drone system implements a swarm of drones. Drones can join or leave the swarm in a dynamic way. A malicious drone could make a spoofing attack and potentially join the swarm. This could allow the malicious drone to eaves drop on the drone-to-drone communication, hijack data traffic in the wireless mesh, and thereby make a man-in-the middle attack. Furthermore, a malicious drone, which has gained access to the multi-drone network will be able to inject spurious/false/corrupted data that potentially will disturb the coordination functions of the swarm. For instance, a formation-flying maneuver could be destroyed or the swarm could be tricked into believe that fellow drones are in false positions. Unfortunately, detecting a malicious drone in an autonomous swarm is a complicated task that may implement sophisticated learning and clustering algorithms [20].

Another spoofing threat arises from a malicious GCS attacking the drone-to-ground interface pretending that it is a legitimate GCS. The malicious GCS could potentially eavesdrop on the telemetry data and/or intercept the C2 channel to take control of one or more drones. Moreover, the malicious GCS would potentially be able to inject false telemetry data into the system tricking the mission supervision function with false status and position data, etc.

Spoofing of the GPS signal may occur when an adversary emits a stronger GNSS/GPS signal with false location information. In [18], the threat to unmanned vehicles, guided by GNSS/GPS sensors to spoofing threats, was analyzed. The findings of this research coupled with the rapid growth of mobile sensor node applications, make it necessary that we begin to address the cyber-physical security challenges that will arise from drone sensor node in use. The authors used a portable civilian GPS spoofer implemented on a digital signal processor (DSP) to characterize spoofing effects. This provided a relative easy attack using a GPS signal-simulator based on a software-defined GPS receiver [19]. The authors of [19] used a manipulated GPS signal, which resembles the original signal, to guiding a drone to the desired location of an adversary and to

land the drone. Furthermore, it is important to note that the spoofing of the GNSS based navigation "is more sinister than intentional jamming because the targeted receiver cannot detect a spoofing attack and so cannot warn users that its navigation solution is untrustworthy" [18].

The above-mentioned spoofing attacks exploit the wireless communication interface. In addition, a network of entities running the Robotic Operating System (ROS) exposes a middleware based communication infrastructure susceptible to cyber-attacks. ROS coordinates communication between the hardware and operational software on the drone by means of a message-passing construct. ROS-based robotic systems comprise "nodes" that communicate by publishing "messages" to different "topics," to which other nodes may listen. The ROS system implements three primary nodes that make the operation of the drone possible. As a matter of fact, recent analyses shows that ROS lacks several security enhancements in order to make it suitable for industrial use [16]**Error! Reference source not found.**. It is noted that under the "restriction of not touching the operating system internals, we cannot easily exclude a publisher or subscriber [in the network], nor can we preclude a new publisher being started and replacing the existing one" [16]. If a malicious ROS node (software) is able to take part in the ROS network it may be able to eavesdrop where the malicious ROS node acts as a subscriber to topics in the network or to inject false information into the network when the malicious node acts as a publisher. False data could fake the control system of the drone to e.g., believe that the battery level is critically low, which would trigger a safety procedure on the drone. In [17], the authors repurposed the mobile sensor node to serve as a surrogate for a car-like robotic system in an effort to emulate the cyber-physical challenges associated with deployed mobile robotics systems based on ROS. The authors demonstrated that spoofing attacks were possible using low-cost, low overhead, cyber-attacks on a robot implementing ROS.

Access to the Cloud Services (CS) is obtained over the Internet, which make the system vulnerable to spoofing attacks on the IP infrastructure. A malicious user could possible run a ROS node over the IP network and eavesdrop communication and/or inject false information in the network. For example, fake waypoints could be sent from the global path planning function sending the drone swarm of to a distant location. Such attack would further allow malicious codes such as viruses, Trojans and bots to run as cloud service functions.

From the taxonomy, we conclude that spoofing attacks are most susceptible for the following elements:

- UAV-SEN
- UAV-COM
- UAV-OS
- GCS-COM
- GCS-OS
- CS-OS
- CS-APP

5.2 Tampering

UAVs are especially vulnerable to physical capture and node tampering attacks [21]. As the drones and possible also the GCS are out in the inspection area with none or limited physical security, they are exposed to tampering attacks that would provide damage on the drones mechanical parts, sensors, power systems, etc. For instance, spraying camera sensors with paint would likely make the useless. Breaking an antenna would significantly inhibit the wireless communication. Since the drones as well as the GCS will be valuable assets for most people, there is furthermore a risk of theft, which may be seen as the ultimate tampering attack.

Suppose a malicious software has got access to the communication network of the UAS, there is a risk that data are compromised. Inspection data (images) can be faked, telemetry data and log files altered as well as data related to the specification of the mission. While corrupting mission data would sabotage the inspection operation it will be relatively easy to detect. In contrast, the corruption of log files could hinder an auditing of an inspection operation. A malicious user would potentially be able to erase any evidence of an intrusion being made. Since data is being exchange between, spread and stored on multiple subsystems i.e., in the drone, in the GCS, and in the CS the malicious users would need to gain access to all relevant subsystem to make the attack hard to detect.

Moreover, a malicious user would be able to tamper with the software running on the drone or the GCS to install malware i.e., viruses, Trojans, key-loggers, botnets etc. or simply delete the running software.

For all data sources there is a risk of data being corrupted in the data storage from a side channels attack by a malicious user e.g., by using a source of electromagnetic radiation or accidentally by a surge of electricity from a power cable. Since the drone and the GCS to a large extend is built from 3rd party suppliers of hardware and software, a vulnerability exists from badly made software and hardware introducing ways to exploit the system. In addition, a malicious manufacturer could implement interception interfaces to eavesdrop on data communication or to allow further malware to be installed.

Another class of tampering attack comes from changes made to the environment. Some algorithms running in the drone supports navigation from recognition of specific features in the landscape. For instance, visual odometry can determine the position and orientation of the drone by analyzing the associated camera images from onboard cameras. Dynamic tampering with the physical environment could disturb the navigation of the drone leading in a wrong direction or potentially tricking a safety procedure.

Concerning communication a tampering attack may arise from the malicious modification of the data passed by a malicious drone when a packet is transmitted along the routing path in the drone swarm [20]. A change in the routing information would possible lead to drones getting their traffic hijacked whereas the dropping of routing information could inhibit data from getting to the GCS, i.e., a black hole attack. Furthermore, replay attacks arising from copying and resending routing information would consume the bandwidth in the network, waste the power of the drone, to achieve the purpose of making the transmission network crash [20].

From the taxonomy, we conclude that spoofing attacks are most susceptible for the following elements:

- UAV-MECH
- UAV-SEN
- UAV-POW
- UAV-APP
- GCS-MECH
- GCS-POW
- GCS-APP
- CS-APP

5.3 Repudiation

Although inspection operations with the D4S aims to be autonomous, there will always be a mission supervisor or a pilot to supervise the operation. A repudiation attack if this user denies having performed certain

interventions or denies to have configured the system in a specific way possible to avoid liabilities or proof of misconduct. For instance, a pilot may attempt to deny that he/she has taken the control of a drone and possible caused an accident of damage to the drone or any civil infrastructure. Furthermore, the pilot may deny that certain safety protocols where not followed.

A drone engineering may attempt to deny that he/she has installed a specific software or made a specific configuration of a drone and in the GCS. This could for instance be a denial of not having run the system with a sufficient degree of security e.g., strong passwords and up-to-date security keys.

In the ROS network, ROS nodes in the drones, the GCS or in the CS may deny that certain information messages was sent or deny have received information that should have been reacted upon.

Black boxes, comprised of a Flight Data Recorder (FDR) component, were introduced in the 1950's and have since become a standard installation on many commercial airliners [22]. Their main purpose was to collect data that could help investigators determine whether an accident was caused by pilot error, air traffic control error, an external event, or airplane system malfunction. The lack of flight data recording could be seen as a way for the system itself to repudiate the course an inspection operation.

From the taxonomy, we conclude that spoofing attacks are most susceptible for the following elements:

- UAV-APP
- GCS-APP
- CS-APP

5.4 Information Disclosure

A malicious user gaining intercepting communication in the UAS will be able to eavesdrop on data exchanges and thereby obtaining knowledge on the how the system is operating such as what control commands are being used, how the communication setup. Obviously, eavesdrop attacks could also give the malicious user access to telemetry data and thereby information about the whereabouts to the multi-drone system as well as access to the inspection payload data. In particular, it is difficult to detect a passive eavesdrop attack as was the case for the US military in Iraq cf. [4]. Passive eavesdrop attacks could be launched from the deployment of malicious drones and/or ground control stations. Since the multi-drone system is connected to an IP infrastructure using ROS middleware, the UAS is vulnerable to information disclosure from malicious remote users intercepting the TCP/IP or the UDP/IP traffic or taking part of the publish/subscriber infrastructure of ROS.

Another threat to information disclose exists for the data storage. In particular, the cloud services (CS) will hold information related to the inspection missions and its progress (e.g., specific drone locations) as well as the inspection images itself. This data or more likely part of these data may temporarily also be cached on the drones themselves or in one or more ground control stations.

Software on a hijacked drone can be copied, read (plaintext) and reverse engineered (binary code). This allows a malicious user to "steal" the knowhow on how the system is being build.

From the taxonomy, we conclude that spoofing attacks are most susceptible for the following elements:

- UAV-COM
- UAV-OS
- UAV-APP

- GCS-COM
- GCS-OS
- GCS-APP
- CS- COM
- CS-OS
- CS-APP

5.5 Denial of Service

Jamming of the communication and scrambling/distortion of signals is a threat to most wirelessly controlled system. Jamming is achieved by introducing a source of noise strong enough to significantly reduce the capacity of the channel. Jamming may be either unintentional or malicious. In addition, we may consider electromagnetic interference from the power line cables is a potential unintentional source of a jamming attack. Jammed segments of bandwidth, once detected, can also be avoided in a spread spectrum scheme. Since jamming is fairly easy to detect and to address, we believe that it can have a low impact on both the user and system.

Jamming of the GNSS/GPS signal will impact on the system navigation possible leading the system to a wrong location or tricking a safety procedure. Although more difficult than jamming, successful GPS spoofing can cause a UAV to go off course, crash, or even be hijacked by an adversary. Spoofing the GPS of a UAV has been proven to be an (effective attack on those using the public GPS system [30].

Another type of denial of service attack arises from the tricking the UAV sensors with false input. For instance, placing obstacles in front of the drone would trick the object avoidance algorithm resulting in a stop of the drone and force it to recalculate its route. Furthermore, vandalism caused to the sensors, such as e.g., spray camera sensors with ink/paint would leave the drone make the drone temporary useless for inspections.

The UAS my furthermore suffer from attacks launched from the Internet such as botnet attacks [28][29]. Botnet attacks can be used to perform distributed denial-of-service (DDoS) attacks, steal data, and inject messages into the network that allow the malicious user to access the device and its connection. As the UAV is essentially a flying computer, software security vulnerabilities such as buffer overflows can apply to a UAV, as well as malware infections. In addition, on-board sensors such as LIDAR and vision-based sensors can be targeted directly with false data injection or jamming.

Similar to threats of WSNs, drones may suffer from the lack of energy to continue its operation. While WSN may suffer sleep deprivation attacks, the multi-drone system may be exposed to a “denial-of-charging” attack. This attack may be intentional or unintentional. If a drone is hindered in sufficient charging it may be rendered useless for further inspection and in worse case getting lost.

From the taxonomy, we conclude that spoofing attacks are most susceptible for the following elements:

- UAV-SEN
- UAV-POW
- UAV-COM
- GCS-COM

- CS-COM

5.6 Elevation of privilege

An unauthorized use may get access to the mission planning and mission control and could potentially be hijacking drones. The malicious user may tune into the drone control frequency and take control of a drone.

Furthermore, malicious software could injects false ROS messages to destabilize the system.

From the taxonomy, we conclude that spoofing attacks are most susceptible for the following elements:

- UAV-OS
- UAV-APP
- GCS-OS
- GCS-APP
- CS-OS
- CS-APP

6 Conclusions of analysis

The results of the Stride Analysis of the D4S use cases and its influence on the multi-drone system show that the different entities/nodes as well as communication between them need to be secure. Table 5 provides a summary of the mapping of threat types to subsystems of the UAS.

Table 5: Mapping of threats to D4S subsystems.

	UAV						GCS					CS			
	MECH	SEN	COM	POW	OS	APP	MECH	COM	POW	OS	APP	INF	COM	OS	APP
Spoofing		X		X	X			X		X					X
Tampering	X	X		X		X	X		X		X				X
Repudiation						X					X				X
Information Disclosure			X		X	X		X		X	X		X	X	X
Denial-of-Service		X	X	X				X					X		
Elevation of Privileges					X	X				X	X			X	X

To counter these threats, the following high-level security requirements need to be further investigated in the design of multi-drone system.

- Access Control
- Device Authenticity and Integrity
- Software and Information integrity
- Mutual Authentication of devices
- Authenticated firmware updates
- Activity Log Files

- Secure Data Storage
- Denial of Service resilience
- Data Chancel authentication, integrity and encryption

7 References

[1] J. Whelan, A. Almehmadi, J. Braverman and K. El-Khatib, "Threat Analysis of a Long Range Autonomous Unmanned Aerial System," 2020 International Conference on Computing and Information Technology (ICCIT-1441), Tabuk, Saudi Arabia, 2020, pp. 1-5. DOI: 10.1109/ICCIT-144147971.2020.9213789

[2] "The STRIDE Threat Model", Microsoft Docs, 2019-12-11, URI: https://msdn.microsoft.com/en-us/library/ee823878%28v=cs.20%29.aspx

[3] "Threat Modeling", Microsoft Docs, 2019-10-07, URI: https://msdn.microsoft.com/en-us/magazine/cc163519.aspx

[4] "US drones hacked by Iraqi insurgents", The Guardian, 2009-12-17. URI: https://www.theguardian.com/world/2009/dec/17/skygrabber-american-drones-hacked . Accessed 2021-04-22

[5] "Computer virus infects drone plane command centre in US", The Guardian, 2011-10-09. URI: https://www.theguardian.com/technology/2011/oct/09/virus-infects-drone-plane-command

[6] Check Point Press Releases, "Check Point Researchers Reported Vulnerabilities in Market-Leading Drone Platform, Enabling Manufacturer to Bolster Security", 2018-11-08, URI: https://www.checkpoint.com/press/2018/check-point-researchers-reported-vulnerabilities-in-market-leading-drone-platform-enabling-manufacturer-to-bolster-security/ . Accessed 2021-05-04.

[7] K. Mansfield, T. Eveleigh, T. H. Holzer, and S. Sarkani. "Unmanned Aerial Vehicle Smart Device Ground Control Station Cyber Security Threat Model", 2013 IEEE International Conference on Technologies for Homeland Security, 2013.

[8] A. Y. Javaid, W. Sun, V. K. Devabhaktuni and M. Alam, "Cyber security threat analysis and modeling of an unmanned aerial vehicle system," 2012 IEEE Conference on Technologies for Homeland Security (HST), 2012, pp. 585-590, doi: 10.1109/THS.2012.6459914.

[9] Kevin Townsend, "Sky-high concerns: Understanding the security threat posed by drones", web blog, 26 September 2019. URI: https://blog.avast.com/what-security-threats-are-posed-by-drones. Accessed 2021-05-13

[10] ETSI, *Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 4; Protocol Framework Definition; Methods and Protocols for Security; Part 1: Threat Analysis*, Technical report no. ETSI TS 102 165-1 V4.1.1 (2003-02), URI: https://ia800701.us.archive.org/18/items/etsi_ts_102_165-1_v04.01.01/ts_10216501v040101p.pdf

[11] D2.4 Use Case Document, Project deliverable, Drones4Safety, Version 3.5, 12 December 2020.¨x

[12] Akyildiz, I. F., & Kasimoglu, I. H. (2004). Wireless sensor and actor networks: research challenges. *Ad hoc networks*, *2*(4), 351-367.

[13] Rappaport, T. S., Annamalai, A., Buehrer, R. M., & Tranter, W. H. (2002). Wireless communications: past events and a future perspective. IEEE communications Magazine, 40(5), 148-161.

[14] D5.1 Specification of the Multi-Drone Swarm System, Project deliverable, Drones4Safety, Version 1.0, 31 March 2021.

[15] ENISA – General threat landscape. URI: https://www.enisa.europa.eu/topics/threat-risk-management/threats-and-trends. Accessed 2021-05-28.

[16] B. Dieber, S. Kacianka, S. Rass and P. Schartner, "Application-level security for ROS-based applications," 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016, pp. 4477-4482, doi: 10.1109/IROS.2016.7759659.

[17] McClean, J., Stull, C., Farrar, C., & Mascarenas, D. (2013, May). A preliminary cyber-physical security assessment of the robot operating system (ros). In Unmanned Systems Technology XV (Vol. 8741, p. 874110). International Society for Optics and Photonics.

[18] Humphreys, TE; Ledvina, BM; Psiaki, ML; O'Hanlon, BW; and Kintner, PM; ssessing the Spoofing Threat:Development of a Portable GPS Civilian Spoofer, in the proceedings of the 2008 ION GNSS ConferenceSavanna, GA, September 16–19, 2008.

[19] N. Shijith, P. Poornachandran, V. G. Sujadevi and M. M. Dharmana, "Spoofing technique to counterfeit the GPS receiver on a drone," 2017 International Conference on Technological Advancements in Power and Energy (TAP Energy), 2017, pp. 1-3, doi: 10.1109/TAPENERGY.2017.8397268.

[20] S. Sun, Z. Ma, L. Liu, H. Gao and J. Peng, "Detection of malicious nodes in drone ad-hoc network based on supervised learning and clustering algorithms," 2020 16th International Conference on Mobility, Sensing and Networking (MSN), 2020, pp. 145-152, doi: 10.1109/MSN50589.2020.00037.

[21] T. Alladi, Naren, G. Bansal, V. Chamola and M. Guizani, "SecAuthUAV: A Novel Authentication Scheme for UAV-Ground Station and UAV-UAV Communication," in IEEE Transactions on Vehicular Technology, vol. 69, no. 12, pp. 15068-15077, Dec. 2020, doi: 10.1109/TVT.2020.3033060.

[22] J. Yapp, R. Seker and R. Babiceanu, "Providing accountability and liability protection for UAV operations beyond visual line of sight," 2018 IEEE Aerospace Conference, 2018, pp. 1-8, doi: 10.1109/AERO.2018.8396532.

[23] Zakeri, R., Shahriari, H. R., Jalili, R., & Sadoddin, R. (2005, September). Modeling TCP/IP networks topology for network vulnerability analysis. In 2nd int. symposium of telecommunications (pp. 653-658).

[24] Software Engineering Institute, Carnegie Mellon University, URI: https://www.sei.cmu.edu/. Accessed 2021-05-31

[25] R. R. Teixeira, I. P. Maurell and P. L. J. Drews, "Security on ROS: analyzing and exploiting vulnerabilities of ROS-based systems," 2020 Latin American Robotics Symposium (LARS), 2020 Brazilian Symposium on Robotics (SBR) and 2020 Workshop on Robotics in Education (WRE), 2020, pp. 1-6, doi: 10.1109/LARS/SBR/WRE51543.2020.9307107.

[26] P. K. Manadhata and J. M. Wing, "An Attack Surface Metric," in IEEE Transactions on Software Engineering, vol. 37, no. 3, pp. 371-386, May-June 2011, doi: 10.1109/TSE.2010.60.

[27] Felix Schuster F, and Holz T, "Towards reducing the attack surface of software backdoors", in Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security (CCS '13), November 2013, pp. 851–862, doi: 10.1145/2508859.2516716

[28] Z. Zhu, G. Lu, Y. Chen, Z. J. Fu, P. Roberts and K. Han, "Botnet Research Survey," 2008 32nd Annual IEEE International Computer Software and Applications Conference, 2008, pp. 967-972, doi: 10.1109/COMPSAC.2008.205.

[29] Reed, T., Geis, J., & Dietrich, S. (2011, August). SkyNET: A 3G-Enabled Mobile Attack Drone and Stealth Botmaster. In WOOT (pp. 28-36).

[30] Arteaga, S. P., Hernández, L. A. M., Pérez, G. S., Orozco, A. L. S., & Villalba, L. J. G. (2019). Analysis of the GPS spoofing vulnerability in the drone 3DR solo. IEEE Access, 7, 51782-51789.

[31] DDS Security, OMG, URI: https://www.omg.org/spec/DDS-SECURITY/1.1/PDF

Annex A: Data from the vulnerability analysis

The table below list the results of the data collection of vulnerabilities on the multi-drone system. It is an unsorted output of input given by members of WP5.

Threat type	Subsystem affected	Description of threat	Threat category
Denial of service	UAV-COM	Jamming of the wireless communication channel. Inhibits or obstrucst coordination between drones	Jamming
Denial of service	GCS-COM	Jamming of the wireless communication channel. Affects the C2 as well as the telemetry channel.	Jamming
Spoofing	GCS-COM	Malicious user is getting access to the wireless communication and send fake telemetry data.	Data corruption
Denial of service	UAV-COM	Jamming of the wireless communication channel. The communication to the cloud backend may be via cellular or via the GCS acting as a relay connecting to the internet using wireless communication	Jamming
Spoofing	UAV	A malicious drone joins the swarm and gets access to data and control information	Eavesdropping
Information disclosure	UAV-COM	A malicious drone or ground station can eavesdrop communication by intercepting wireless signals and/or ROS communication.	Eavesdropping
Spoofing	UAV-SEN	Spoofing can fool the drones into a false position.	Signal scrambling/distortion
Denial of service	UAV-SEN	Jamming of GNSS signal can make a drone inoperable.	Jamming
Denial of service	CS	Cloud services are unresponsive due to Distributed Denial of Service (DDoS)	Loss of availability
Information disclosure	CS-APP	Data stored in cloud serviced is compromised (Inspection results, drone locations, mission history...)	Data corruption
Elevation of privileges	CS-COM	Unauthorized access to mission planning (hijacking drones)	Loss of availability
Tampering	CS-APP	Tampering the inspection result data on the cloud	Data corruption
Tampering	CS-APP	Tampering the mission data on the cloud	Data corruption
Tampering	UAV-APP	Altering data stores in the drone by accessing the storage or through interaction with an electromagnetic radiation source	Data corruption
Tampering	UAV-APP	Installing malicious software in the drone.	Malicious code
Denial of service	UAV-COM	Flood the network with unwanted traffic such as wireless packets, SYN messages (TCP), UDP strem, ICMP (Ping) or similar.	Loss of availability
Elevation of privileges	UAV	Hijacking of drone. The malicious user may tune into the drone control frequency and take control of a drone.	Loss of availability
Tampering	UAV-SEN	A supply chain threat exists because drones are largely manufactured from components manufactured abroad where a "back-door" function can be implemented.	Eavesdropping

Tampering	UAV	Capture drone and learn how the drone hardware and software is build.	Social or reverse engineering
Tampering	GCS	Capture a ground control station and learn how the drone hardware and software is build.	Social or reverse engineering
Tampering	UAV-MECH	Capture drone and break it or parts of it.	Vandalism
Spoofing	GCS-COM	Take control of a drone with an unauthenticated wireless controler.	Theft
Spoofing	UAV-COM	Unauthorized or malicious ROS node gets access to the ROS communication infrastructure	Eavesdropping
Elevation of privileges	UAV-COM	Malicous software injects false ROS messages to destabilize the system.	Signal scrambling/distortion
Denial of service	UAV-SEN	Strong electrical interference causes a drone to become instable	Signal scrambling/distortion
Repudiation	UAS	A pilot/mission operator deny that a certain operation (e.g., flying path) has been taken.	Repudiation
Information disclosure	UAV-APP	Software on a hijacked drone can be copied, read (plaintext) and reverse engineered (binary code).	Social or reverse engineering
Repudiation	UAS	A pilot denies that a control action hand been taken by the pilot.	Repudiation
Information disclosure	GCS-APP	Access mission data on the GCS.	Eavesdropping
Tampering	GCS-APP	Alter mission data on the GCS.	Data corruption
Tampering	UAV-APP	Send fake inspection results to Ground Control Station.	Signal scrambling/distortion
Tampering	CS-APP	Alter digital map for path planing (Bridges, Railways, No-flight-zones, …)	Data corruption
Spoofing	UAV-SEN	Alter on board drone data from flight controller (for example battery status)	Signal scrambling/distortion
Spoofing	GCS-COM	Add a unauthorized GCS to the wireless mesh network to eavesdrop on the communication to and from the drones.	Eavesdropping
Denial of service	UAV-SEN	Disturb visual sensors e.g., with obstackles, smoke etc. to interfere with the drone navigation.	Signal scrambling/distortion
Denial of service	UAV-SEN	Disturb visual sensors e.g., with obstackles, smoke etc. to interfere with the collision avoidance.	Loss of availability
Denial of service	UAV-APP	Run an virus on the drone which disables the other software	Loss of availability
Tampering	UAV-OS	Delete the software on the drone	Vandalism
Tampering	UAV-SEN	Alter or fake visual sensor date to force a crash.	Signal scrambling/distortion
Information disclosure	GCS-APP	Installation of malwares, trojans, key-loggers in the GCS to disclose information	Eavesdropping

Spoofing	CS-OS	An unauthorized ROS node provides a malicious cloud service by evesdropping in the ROS network and/or injecting false information.	Eavesdropping
Information disclosure	CS-APP	Installation of malwares, trojans, key-loggers in the GCS to disclose information	Eavesdropping
Tampering	UAV	The drone is captured and stolen. Or the drone accidentially gets lost of destroyed in a crash	Theft
Tampering	GCS	The GCS is stolen.	Theft
Tampering	UAV-COM	Modifying routing protocol messages in the network to disconnect drones from the swarm or create a black hole attack for the GCS	Data corruption
Repudiation	UAV-APP	A drone engineer refuses that he/she have installed specific software or made specific configurations of the drone	Repudiation
Repudiation	GCS-APP	A drone engineer refuses that he/she have installed specific software or made specific configurations of the drone	Repudiation
Repudiation	UAS	ROS nodes may deny that certain ROS messages, leading to a system breach, was sent.	Repudiation
Repudiation	UAV	The UAV system could in a sense "forget" or "omit" to record its operation and in that sense repudiate that a specific flight was conducted	Repudiation