# Methodology for risk management related to cyber-security of Unmanned Aircraft Systems

Trung Duc Tran, Jean-Marc Thiriet, Nicolas Marchand, Amin El Mrabti, Gabriele Luculli

# Methodology for risk management related to cyber-security of Unmanned Aircraft Systems

Trung Duc TRAN*,†, Jean-Marc THIRIET*, Nicolas MARCHAND*, Amin EL MRABTI† and Gabriele LUCULLI†
*Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, 38000 Grenoble, France
Email: Trung-Duc.Tran@gipsa-lab.grenoble-inp.fr
†SOGILIS Company, 38000 Grenoble, France
Email: amin@sogilis.com

***Abstract*—Nowadays, the market of Unmanned Aerial Systems (UAS) develops progressively with a lot of applications such as: infrastructure monitoring, law enforcement, environment research, goods transportation, etc. However, due to the lack of human observation, communication capacities, and protection, a UAS is an appropriate target for a criminal or a terrorist cyber attack. Therefore, risks related to the cyber-security need to be taken into account during the development of a UAS. This document presents a risk management methodology related to cyber-security. The expected result of the application of this methodology is a list of cyber-security requirements which guides the development of cyber-security countermeasures.**

***Index Terms*—Unmanned Aircraft System (UAS), cyber security, risk management, cyber-security requirement**

## I. Introduction

Nowadays, Unmanned Aerial Systems (UAS) have a wide range of critical applications such as air intelligence, infrastructure monitoring, law enforcement, goods transportation, etc. Actually, the safety and cyber-security levels of most UASs do not correspond to their targeted usage. In order to make UAS reach the safety level of a certificated aerospace system, Sogilis company developed a software development process according to the DO-178C standard (Software Considerations in Airborne Systems and Equipment Certification). In this study, we aim to add the cyber-security aspect to the development process. The output of the cyber-security analysis is used as the input of the development process. Our general approach is presented in Figure 1.

In general, people in charge of security have to answer some questions: "Have all the threats been identified?" or "are these countermeasures all necessary?" [1]. In order to answer these questions, the risk management could provide a systematical and effective way to detect, analyze, evaluate possible security loss and select adequate countermeasures to mitigate its impact. Moreover, risk management could allow security manager to balance operation vs. economic cost of implementing security countermeasures [2].

To the best of our knowledge, most research related to security of UAS focus on discovering and solving the individual security issue such as GPS spoofing [3], [4], secure communication [5], [6], sensor spoofing [7], [8], virus [9]. Through this study, we aim to treat the cyber-security issues of UAS in the global view by developing a risk management methodology. The expected result of the application of this methodology is a list of cyber-security requirements which will be used to define the system requirements in the development process.

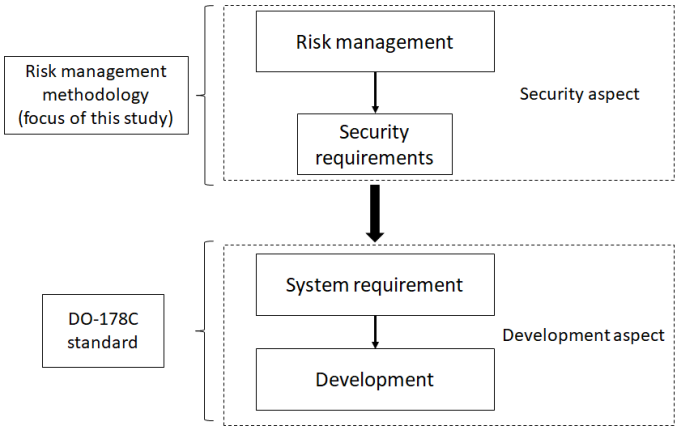


Fig. 1. General approach

The remaining of this document is organized as follows. The background and related work on risk management in both industry and research is presented in Section II. The proposed methodology is then given in Section III. A study case and the result of the application of the methodology is given in Section IV. We conclude our works and present our perspective on the future works in Section V.

## II. Background and related works

### A. Incidents and researches related to cyber-security of UAS

Actually UASs are exposed to several cyber security risks. There are many reports on cyber-security breach and researches on attack simulation that show the importance of cyber-security in the UAS development.

A GPS jamming attack was claimed as the cause of the crash of S-100 Camcopter, a rotor based UAV, resulting in a dead and two injured in 2012 [10]. Another attack related to GPS spoofing was suspected of being executed in 2011, resulting in the capture of RQ-170, a military UAS [11]. Several cases of security violation by video interception on military are also reported in the literature [12], [13]. It is quite surprising that

an off the shelf product of 26 dollars could be used to intercept video data from a military UAS [13].

There are also a lot of research works on simulated cyber-security attacks [8], [14]–[16]. These are especially important because we can gain new knowledge about potential cyber-security breaches. Therefore, we can reduce the security risk on the UAV. Vattapparamban et al experimented several hijack attacks by exploiting the vulnerabilities of Wifi communication of three small unmanned aircraft vehicles (UAVs) [14]. Heiges et al simulated two attack scenarios on a UAV [15]. The first one is the malicious change of its flight plan, the second one is the corruption of the on-board camera. Both were executed by injecting fake commands into the autopilot. Son et al succeeded to crash a drone by manipulating the normal gyroscope behaviour. It was disturbed by sound waves at the right resonant frequency which caused an unexpected behaviour of the autopilot and therefore the drone crashed [16]. Davidson et al argued that the optical flow sensors used for navigation represent a vector for adversarial control [8]. In order to demonstrate this argument, the authors experimented on a real UAV.

To ensure the global cyber-security of UAS is a very challenging task. We reported a few examples of cyber-security risks and their related attack methods but, unfortunately, they evolve day by day. Therefore it is quite important to develop new methodologies in order to improve the level of UAV cyber-security even in the case of unknown or partially unexpected attacks.

### *B. Related work in risk management*

A risk management is a systematic and effective process to deal with security risks. It could contain different activities: risk identification, risk assessment, decision making for control the risks. In this section, we present different methodologies, methods are currently adopted in industry for the risk management. These methodologies/methods are used for varied systems which are more or less similar to UAS such as automobile, avionic, smart medical system and information system.

**ISO 27005 standard** was first considered as a guidance to information security risk management in an organization [17]. Beyond the boundary of information security in an organization, ISO/IEC 27005 is also referred as a guidance to cyber security in cyber physical systems. This standard constitutes a framework for risk management process rather than providing a specific risk management method [1]. The framework of ISO27005 is shown in Figure 2.

**Failure Mode Vulnerability and Effect Analysis (FMVEA)** [18] is a risk assessment for both security and dependability, which is extended from Failure Mode Effect Analysis (FMEA). In order to identify the security risks, the author firstly identifies the possible attacks against components of protected system then define the impact of the component failures on the system. But the author does not detail how the attacks are identified.

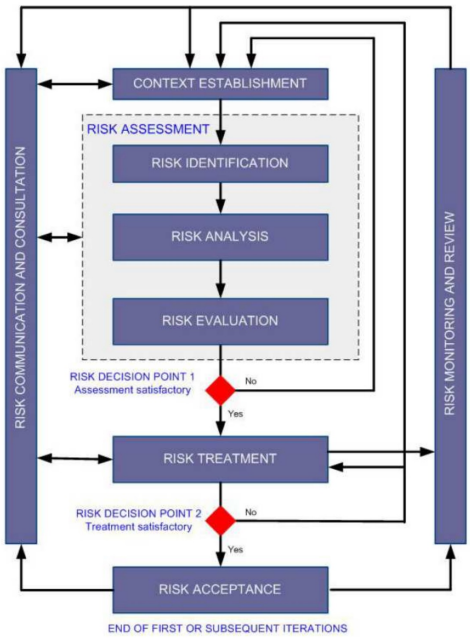


Fig. 2. An information security risk management process from ISO/IEC 27005:2011 [17]

**Attack tree** [19] is an inductive risk assessment method. This method is commonly used in different security domains (automobile [20], smart health [21], industrial control system (ICS) [22]). The attack tree method describes the attacks on a system by first identifying the goal of the attack as a root node of a tree. The way the adversary reaches this goal interactively and incrementally is expressed as child nodes in the tree. Each path in the attack tree from the leaf node (or a set of leaf nodes) to the root node describes a unique attack on the system [21].

**E-safety Vehicle Intrusion proTected Applications (EVITA)** [23] [20] is a research program in automotive industry. This project presented a risk management methodology dedicated automobile on-board network. The aim of this methodology is to define a list of security requirement to design and verify security solutions, which is similar to our purpose.

**Method for Harmonized Analysis of Risk (MEHARI)** is a risk management methodology for information security of an organization. In this methodology, the risks are reasoned based on analyzing the possible malfunction of different functionalities, then they are evaluated and treated based on the supporting tools of MEHARI. Because MEHARI is designed to control the information security risks of an organization, this is not suitable for other domain such as security of UAS. However the strategy to identify the risks could be useful for UAS risk management.

**ED202A/DO326** is a guidance for airworthiness security process. The objective of this document is to add new processes to handle the risk of intentional unauthorized electronic to the current process of aircraft development and certification [24]. For this purpose, this document provides a security

risk management methodology compliant with ISO 27005 standard.

**Specific Operation Risk Assessment (SORA)** is a risk assessment methodology dedicated to UAS application. This methodology focuses on evaluating only the safety level of an operation. This factor is qualitatively estimated based on the likelihood that the occurrence "Out of control of operation" could lead to injuries of people on ground, collisions with other aircrafts, and damage infrastructures on ground. Based on this factor, the applicant could chose appropriate mitigation solutions with their level of robustness.

Through our study in the state of the art, we find that it lacks a risk management methodology for the UAS development. The methodologies the closest to our application are ED202A/DO326A and SORA. Designed as a part of the civil aircraft development, ED202A /DO326A is too complex to be integrated into our development process while SORA focuses only on evaluating the safety of a UAS operation. Therefore, this research is intended to propose a risk management methodology which is dedicated to UAS security, simple enough to implement and capable to ensure more or less the completeness of the protection.

## III. Proposed methodology for risk management of UAS

By inspiring methodologies existing in other domains, especially the ones listed in the last section, we developed a simple risk management methodology dedicated to UAS development. The methodology includes four principal activities: "Context establishment", "Risk identification", "Risk analysis and evaluation" and "Treatment" as shown in Figure 3. In the Context establishment activity, we propose a method to collect and arrange all the information on the situation, on the protected system, which define the scope of risk management. For the Risk identification activity, we propose a method to identify the possible security risks based on the attack tree method and the malfunction analysis. The Risk analysis and evaluation activity is to define the priority of each defined risk. The risk with the highest priority needs to be treated first with robust solutions. The last one, Treatment is to define the security requirements, which are used to design, verify security solutions. These activities are described with more details in the remaining of this section.

### A. Context establishment

In this methodology, the context establishment (activity A) aims to prepare necessary information for the next analysis in the risk assessment activity. This activity includes the following steps:

- Operation description
- System under consideration description

*1) Operation description:* This step aims to describe as detailed as possible the objective and process which the deployed system needs to achieve in the point of view of operators. For this purpose, we utilize the guideline of JARUS-SORA [25] for collecting and presenting operation information as follows:

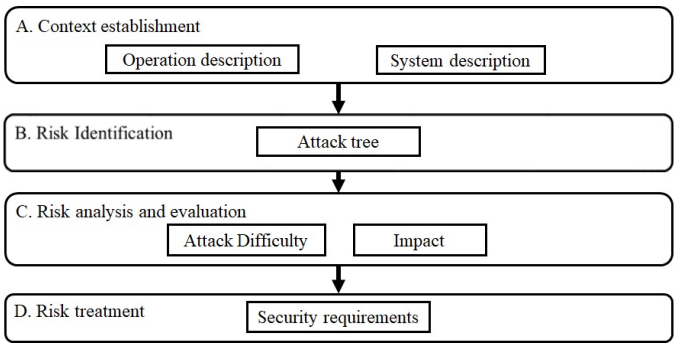


Fig. 3. Work-flow of the proposed methodology

- Detailed description containing all information to get understanding of how, where, and under which limitations or conditions the drone is deployed.
- Detailed description containing all information about the type of operation such as (Visual Line Of Sight or Beyond Line Of Sight) and the level of involvement of operators or levels of automation of the drone during the flight.
- Detailed description about the processes of system deployment and maintenance as well as the people involved in these processes
- Detailed description about contingency procedures in place for any malfunctions or emergency cases (e.g. run out of battery, loss of connection).
- Several cyber-security assumptions about the system under consideration, the environment that allows to reduce the scope of analysis and neglect several kinds of attacks. For instance, an assumption could be that all staffs are trusted so that all attacks launched intentionally by internal staffs is neglected.

Note 1: From this description, several informations could be extracted such as functions that the system needs to be achieved (e.g. following a specific trajectory, sending video back to the Ground Control Station - GCS), reference factors used to analyze the severity of impact (e.g. number of deaths in case of the operation that a drone flies over a crowd, financial loss in case of operation that a drone transports goods).

Note 2: The cyber-security assumptions need to be identified carefully; if not, potential attacks could be neglected.

*2) System under consideration description:* The purpose of this step is to obtain the necessary knowledge about the protected system. This step focus on collecting several kinds of information: architecture, cyber-security environment, interface, function.

- **Architecture**: a system could be decomposed into small components. These elements and their interconnections should be identified.
- **Environment**: all people, external systems which could interact with system under consideration. For instance,

in case of a UAS, cyber-security environment could be maintenance personnel, manufacture, Internet, operators, etc. For each element of environment, their capabilities of access and roles need to be detailed.

- **Interface**: all entry points that elements of environment could interact with the system. For example, in case of a drone, the ground control station sends command data to the drone via RF communication, therefore the RF communication is an interface of the drone.
- **Functions**: all discrete actions (described by action verbs) necessary to achieve the systems objectives. The information on **system functions** could be deduced from the system operation information. The information on the component functions could be deduced from its output which is presented in the architecture. For example in case of a drone, the system function could be following a pre-determined trajectory, recording and sending video back to the ground station. In the function description, it should also detail requirements for this function. For example, for "sending video back to the ground station" function, it should detail intended quality of video, confidentiality of video data, etc.

A UAS is a complex system combined by many components such as autopilot, GPS, RF module, camera. Each component has also its own architecture, function, interface, environment. Therefore, all mentioned information should be collected in many abstract levels. For example, beside of architecture, interface, function, cyber-security environment of the UAS, we need to know the ones of autopilot, RF module, camera, etc.

Note: Depending on the process of development (design, test, documentation) and the status of the system (under development or ready to use), these information could exist (documented) or not. In case they do not exist, they should be deduced from existing information in the way that ensures the completeness and the logic of information.

### *B. Risk identification*

In the risk identification step, we aims to achieve three objectives. The first objective is to identify as exhaustively as possible risks. The second one is to bring to light the nature of the risks and their evolution (including basic action of attackers, malfunctions in components at different abstract levels and a malfunction at the system level). The last one is to facilitate the security requirement selection. For this purpose, this methodology adopts a new version of the attack tree for this step. The process for building attack trees is shown in Figure 4.

Firstly, each attack tree starts with a malfunction of the system at the highest abstract level, as a root node that presents the goal of attack (for example, "crash of drone" or "disclosure of the observation video"). These malfunctions could be directly deduced from desired functions of system identified in the context establishment. Each malfunction is considered as the loss of one of three security attributes (Confidentiality, Integrity, Availability) see Table I.

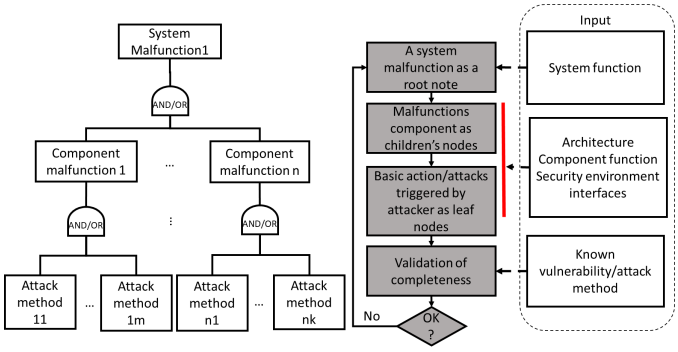


Fig. 4. Attack tree construction work-flow

TABLE I
MALFUNCTIONS DUE TO LOSS OF SECURITY ATTRIBUTES

| Security attributes | Malfunction description |
|---|---|
| Availability | Malfunction presenting the denial of access to the function |
| Integrity | Malfunction presenting the misuse or the in-correction of the function |
| Confidentiality | Malfunction presenting the disclosure of information/data |

*For example, in regard to "fly and follow a pre-determined trajectory" function, we can identify two malfunctions of drone: crash (loss of availability) and divergence from pre-determined trajectory - following the trajectory defined by attackers (loss of integrity).*

Next, malfunctions in components contributing to root malfunction is deduced and expressed as child nodes of the attack tree that presents sub goals of the attack (e.g. the autopilot provide incorrect command to the motors). In order to deduce the malfunction of components, we use lists of component functions and architectures as inputs.

*For example, "crash of drone" is a malfunction at the highest level and assigned as a root node. This malfunction is related to "fly following a pre-determined trajectory" function. This system function is achieved by several component functions:*

- *Autopilot: provide motor command*
- *GPS: providing position data*
- *Inertial measurement unit (IMU): providing attitude data.*

*Therefore the child nodes of "crash of drone" could be "GPS provide incorrect position data", "GPS unable to provide position data", "autopilot unable to control aircraft", etc.*

Then this process is repeated for identifying causes of the malfunction of the sub-systems until reaching the lowest level elements (where information is available). Lastly, the attack tree ends with leaf nodes expressing malicious actions or attack methods that could be launched by an attacker for triggering attacks. These malicious actions could be deduced from information about interface, environment.

After being finished, the attack trees give us a visual presentation about the whole of risks related to a system

function. Each path from leaf node to root node expresses an attack scenario that attackers could carry out. Each attack scenario is a cyber-security risk which we need to be evaluated in the next steps. Because the process of building attack trees is deductive, the result is more or less influenced by the capacity of the person who performs the analysis. Therefore, at the end, the completeness of result needs to be verified by checking if all documented attack methods have been identified in the attack trees.

Note: During deduction process, however some malfunctions/vulnerabilities are considered as very difficult to occur, they should be kept on the attack tree if the link between them and higher malfunctions/ malicious action is logic. For example, "flashing GPS with malware via its USB port" is difficult to occur but could happen, so that it needs to be shown in the attack tree.

### C. Risk analysis and evaluation

This step aims to determine which attack scenario needs to be considered and which one could be neglected. The basic idea in this step is similar to the one in safety analysis where the level risk is characterized by two factors: the likelihood and the severity of impact. However, the likelihood of an attack is difficult to determine due to the lack of feedback. Instead of likelihood, we evaluate the difficulty of attack (DOA), which express the total effort an attacker needs to carry out a successful attack. The attacks which are easy to perform but could give a major impact, should be treated first. The attacks which are difficult to perform and have a minor impact, could be neglected or treated with low priority. Table II shows the mechanics used to decide the risk level of each attack scenario (L, M, H denote representatively Low, Medium and High risk level).

TABLE II
RISK LEVEL

| | | | | | | |
|---|---|---|---|---|---|---|
| DOA | None | L | M | M | H | H |
| | Basic | L | L | M | M | H |
| | Moderate | L | L | L | M | M |
| | High | L | L | L | L | M |
| | Very High | L | L | L | L | L |
| | | No Impact | Low | Medium | Strong | Very Strong |
| | Severity of attack | | | | | |

In this methodology, the difficulty of each attack and the severity of impacts are evaluated qualitatively by more than one person. The severity of attack could be reasoned from operation information collected in the context establishment activity. The difficulty of an attack could be evaluated based on the nature of necessary equipment (e.g. cheap or expensive, popular or not), the necessary knowledge of attack techniques and systems to carry out the attack.

In the context of aerospace industry, we adopt the attack difficulty scales proposed in ED203 [26] - the guideline document of ED202A/DO326. The difficulty of each attack is firstly measured in range from 0 to 30 points then is represented in 5 level scale (None, Basis, Moderate, High and very High) as shown in Table III.

TABLE III
DIFFICULTY OF ATTACK SCALE

| From 0 to 6 | From 7 to 12 | From 13 to 18 | From 19 to 24 | From 25 to 30 |
|---|---|---|---|---|
| None | Basic | Moderate | High | Very High |

The difficulty of attack is determined by three criteria: "Preparation Means", "Execution Mean" and "Windows of Opportunity". The final difficulty point of each attack is equal to the sum of points for these criteria. Tables IV, V, VI are used as evaluation tools.

TABLE IV
PREPARATION MEANS

| | | Knowledge | | |
|---|---|---|---|---|
| Equipment | | None/Public information and no preparation time | Uncontrolled information and no signification preparation time | Insider Knowledge or Significant preparation time |
| | None/Standard[1] | 0 | 2 | 6 |
| | Special COTS[2] | 0 | 2 | 6 |
| | Special[3] | N/A | 4 | 6 |
| | Bespoke[4] | N/A | 5 | 6 |

TABLE V
WINDOWS OF OPPORTUNITY

| Points | Description |
|---|---|
| 0 | The attack can be carried out at any time |
| 1 | The attack can be carried out during regular cruise flight. |
| 2 | The attack vector is available while the aircraft is on the ground. |
| 3 | Maximum effectiveness for mandatory operational procedures limiting the window of opportunity. |
| 6 | The attack vector is only available in a restricted time phase, e.g. on the ground in maintenance mode |
| 8 | The attack can only be carried out during a very restricted time slot independent from the flight phase (e.g. during system reboot). |

TABLE VI
EXECUTION MEANS

| | | Expertise | | | |
|---|---|---|---|---|---|
| Equipment | | Layman | Proficient | Expert | Multi Expert |
| | None/Standard | 0 | 2 | 6 | 10 |
| | Special COTS | 0 | 2 | 6 | 10 |
| | Special | N/A | 4 | 6 | 12 |
| | Bespoke | N/A | 5 | 6 | 12 |

This analysis is qualitative and strongly based on the experience/knowledge of the applicants. We know that this can be somehow a limitation of our approach because quantitative assessments are to be preferred from the engineer perspective. We hope to be able to improve this in the future.

### D. Treatment

For each threat scenario selected for treating in the previous step, a set of cyber-security requirements should be established. A cyber-security requirement is not a specific security

[1]No equipment or something commonly already found

[2]Something which can be readily bought, but which is usually not yet in the possession of an average person

[3]Something which cannot be readily bought, but which needs to be assembled/built

[4]Special equipment which requires a bit amount of resources to assemble

measure, but it is only a security objective that needs to be fulfilled to ensure the cyber-security of the system. For each cyber-security requirement, one or more security measures could be considered. They need to be tested/simulated and evaluated (cost, effectiveness) before being selected for wiring down system requirements.

In this methodology, we adopt the classification of security requirement mentioned in ED202A/DO326A [26] as follows:

- Preventive: The aim is to discourage a malicious user from causing a malfunction
- Deterrent: The aim is to prevent an occurrence of a malfunction
- Detective: The aim is to detect and report a malfunction or malicious action of an attacker.
- Corrective: the aim is to react to a malfunction when it occurs
- Restorative: the aim is to put the system back to the normal status after a malfunction

## IV. Case study

In this section, we present the application of our methodology for a case study, in which a UAS is used to observe a highway in auto flight mode. The video captured by the UAV and the flight information are sent to ground and displayed to operators on the screens of Ground Control Station (GCS) computers. During the operation, the UAV will fly and follow a pre-defined trajectory alongside the highway. From the start to the end of the flight, the UAV flies all the time in automatic mode under Beyond-line-of-sight (BLOS) observation of operators. The operators could use three simple commands: start the flight, end the flight (back to stand-by mode) and go home. The architecture of this UAS is shown in Figure 5.

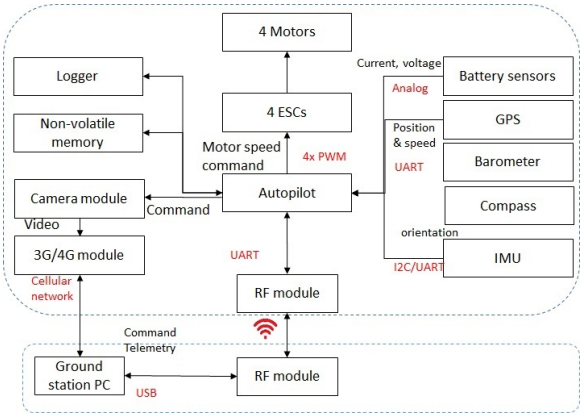


Fig. 5. Architecture of a UAS.

From the description of the system operation, we defined four system functions which need to be protected:

- **Function 1: Fly the vehicle following automatically a pre-determined trajectory**: The drone must follow a flight plan predetermined by manufacturer and embedded in the autopilot. A flight plan contains several way-points. Each way-point contains information on coordinates, altitude in reference to sea level or ground level.
- **Function 2: Provide flight information to operators**: all status informations such as attitude, position, predetermined trajectory, battery information will be sent to the ground and displayed on the screen of the GCS computer. Only operators in charge have right to access these informations.
- **Function 3: Provide observation video to Operators**: the video captured by camera is sent to the ground and displayed on the screen of GCS computer. Only operators in charge have right to access these informations.

Based on the system functions above, we define malfunctions which the attacker want to trigger. Each malfunction is related to the loss of one cyber-security attribute (integrity, avaibility, confidentiality) of a system function. The list of malfunctions is presented as follows:

- **Malfunction 1_Availability** - Crash of UAV: Due to malicious action, the UAV losses its attitude and crashes. Because of flying over a highway, the crash of the UAV could cause a lethal accident.
- **Malfunction 1_Integrity** - Deviation from trajectory: Under an attack, the UAV deviates from its trajectory and flies following the trajectory defined by attacker.
- **Malfunction 1_Confidentiality** - No relevant.
- **Malfunction 2_Availability** - Unavailability of flight information: Under an attack, the flight information is no more available and the operators could not recognize the situation. This malfunction could help attacker launch other attacks or makes the operation be canceled.
- **Malfunction 2_Integrity** - Fake flight information: Under an attack, the fake flight information is provided to operators which makes them make incorrect decisions.
- **Malfunction 2_Confidentiality** - Disclosure of flight information: Under an attack, the attacker could gain unauthorized access to the flight information, which could help the attacker launch other attacks.
- **Malfunction 3_Availability** - Unavailability of video: Under an attack, the operators could not access to the observation video.
- **Malfunction 3_Integrity** - Fake video: Under an attack, the operators receive the fake observation video made by attacker. This malfunction do not impact directly the safety of the operation, but it makes the objective of operation totally failed.
- **Malfunction 3_Confidentiality** - Disclosure of video: Under an attack, the attacker could gain unauthorized access to observation video which impacts on the privacy of the observed people.

For each aforementioned malfunction, we build an attack tree, in which the malfunctions of system are decomposed into the malfunctions of components. The ADTool [27] was used to draw attack-trees. For example, the malfunction 2_Integrity “Fake flight information” attack tree is presented in Figure 6.

Through the attack tree ”Fake flight information”, we define four possible attack scenarios. Because the fake flight infor-

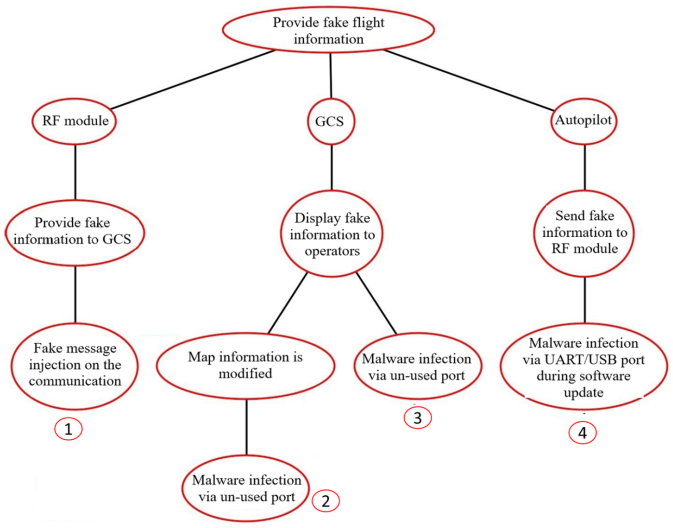


Fig. 6. The complete attack tree expressing the ways of which attacker deceive operator with the fake information - Malfunction 2_Integrity

TABLE VII

| Malfunction | Attack senario | DOA | | | | Severity | Level |
|---|---|---|---|---|---|---|---|
| | | Prepare | Opportunity | Execution | Total | | |
| Provide fake flight information | 1 | 2 | 1 | 2 | None (4) | Strong | High |
| | 2 | 4 | 3 | 6 | Moderate (13) | Strong | Medium |
| | 3 | 4 | 3 | 6 | Moderate (13) | Strong | Medium |
| | 4 | 4 | 6 | 6 | Moderate (16) | Strong | Medium |

mation could lead to serious accidents, we consider that the severity of the defined attack scenarios is at "'strong" level. The evaluation of these attack scenarios is shown in Table VII.

Based on the nature of the nodes of the attack tree, we define different cyber-security requirements as shown in Figure 7. These requirements will be implemented in order of risk level of related attack scenarios.

- Requirement 7: During autopilot software update, the autopilot shall verify the integrity of firmware.
- Requirement 8: Autopilot shall allow only manufacture to flash its memory and shall verify the authentication of manufacture identification.
- Requirement 14: Whenever receiving a packet, the RF module shall verify that this packet is original from one of its paired RF module. The fake packet shall be rejected.
- Requirement 15: Whenever receiving a packet, the RF module shall verify the integrity of the packet in term of time, payload and the origin.
- Requirement 20: Whenever receiving a packet from RF module, the GCS computer shall verify that this packet is generated by its autopilot.
- Requirement 21: The GCS shall verify and ensure the integrity of the map and the flight plan store in GCS computer.
- Requirement 24: The access to the connection port of GCS computer should be limited to only the authorized people by using both software and hardware solutions.

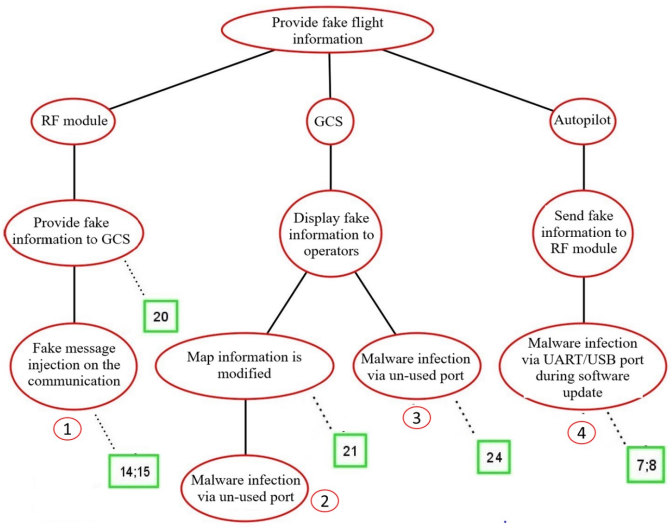


Fig. 7. "Fake information to ground" attack tree with security requirements

Totally, for this study case, we build 6 attack trees. In these attack trees, we list all possible attack methods (leaf nodes) mentioned in the literature of UAS (real attack, test, or simulation). However, it is not enough to make our analysis exhaustive, because the information on real/new attack methods is rarely published. In order to ensure the coverage of all risks, instead of attack methods, we focus on identifying exhaustively malfunction of system/component which could be deduced from the available information on our system. Once the cyber-security requirements for a malfunction is deployed, we could protect the system against not only identified attack methods but also some unknown related to this malfunction. For example, there could be more than one attack method which could lead to the malfunction "RF module provide fake information to GCS". By introducing the cyber-security requirement 20, we could cover also the unknown attack methods that lead to this malfunction (see Figure 7).

As a result, at the end, we establish totally 24 cyber-security requirements. For each path from a leaf node to the root of attack trees, we always have at least one cyber-security requirement. Therefore, all risks identified in our attack trees are taken into account. However, most of the requirements are detective. In order to strengthen the cyber-security of the system, the cyber-security requirements in other class (corrective, restorative) should be added.

## V. Conclusion

In this work, we develop a methodology of risk management in order to integrate a cyber-security approach into the UAS development process of Sogilis Company. Our methodology consists of four activities: collecting the knowledge in the system, identifying risks, judging risk priorities, and establish cyber-security requirements. In order to illustrate our methodology, we present a study-case based on a UAS

Presently, our methodology has both advantages and limitations. An advantage is that it helps improve the completeness of the protection. Traditionally, we needs to identify exhaustively the ways that an attacker could penalize the system, however the task is impossible because the attack method always evolves and the result depends on the capacity of the analysts. Therefore, it's difficult to protect the system from the new/unknown attacks. Instead of that, we focus on identifying exhaustively the possible malfunctions of system/component triggered by an attacker. This task is feasible because the malfunctions could be deduced from the nature of the system which is stable and well documented. The other advantage is that this work is dedicated to UAS application. The limitation of this work is that it could only take into account the cyber-security after the operation and system are defined. We believe that the cyber-security risks should be also considered soon in the phase of concept of operations (CONOPS) and initial system description

As a part of the future work, firstly we attempt to extend our methodology in order to take into account the cyber-security in CONOPS and initial system description. Secondly, we attempt to fulfill the gaps between our cyber-security requirements and the actual development process so that the developers can implement and verify solutions exactly.

## REFERENCES


[1] *MEHARI Overview*, Club de la sécurité de l'information français (CLUSIF), Apr. 2010.

[2] Michael E.Whitman and Herbert J.Mattord, *Principles of information security*, 4th ed. Course Technology, 2012.

[3] Q. Zou, S. Huang, F. Lin, and M. Cong, "Detection of gps spoofing based on uav model estimation," in *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, Oct 2016, pp. 6097–6102.

[4] Y. Qiao, Y. Zhang, and X. Du, "A vision-based gps-spoofing detection method for small uavs," in *2017 13th International Conference on Computational Intelligence and Security (CIS)*, Dec 2017, pp. 312–316.

[5] M. Haluza and J. echk, "Analysis and decoding of radio signals for remote control of drones," in *2016 New Trends in Signal Processing (NTSP)*, Oct 2016, pp. 1–5.

[6] J. Bian, R. Seker, and M. Xie, "A secure communication framework for large-scale unmanned aircraft systems," in *2013 Integrated Communications, Navigation and Surveillance Conference (ICNS)*, April 2013, pp. 1–12.

[7] W. Chen, Z. Duan, and Y. Dong, "False data injection on ekf-based navigation control," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, June 2017, pp. 1608–1617.

[8] D. Davidson, H. Wu, R. Jellinek, V. Singh, and T. Ristenpart, "Controlling uavs with sensor input spoofing attacks," in *10th USENIX Workshop on Offensive Technologies (WOOT 16)*. Austin, TX: USENIX Association, 2016. [Online]. Available: https://www.usenix.org/conference/woot16/workshop-program/presentation/davidson

[9] K. Mansfield, T. Eveleigh, T. H. Holzer, and S. Sarkani, "Unmanned aerial vehicle smart device ground control station cyber security threat model," in *2013 IEEE International Conference on Technologies for Homeland Security (HST)*, Nov 2013, pp. 722–728.

[10] Y.-S. Lee, Y.-J. Kang, S.-G. Lee, H. Lee, and Y. Ryu, "An overview of unmanned aerial vehicle: Cyber security perspective," pp. 128–131, 08 2016.

[11] E. Yadereli, C. Gemci, and A. Z. Akta, "A study on cyber-security of autonomous and unmanned vehicles," *The Journal of Defense Modeling and Simulation*, vol. 12, no. 4, pp. 369–381, 2015. [Online]. Available: https://doi.org/10.1177/1548512915575803

[12] C. G. L. Krishna and R. R. Murphy, "A review on cybersecurity vulnerabilities for unmanned aerial vehicles," in *International Symposium on Safety, Security and Rescue Robotics (SSRR)*. IEEE, Oct. 2017, pp. 194–199. [Online]. Available: http://ieeexplore.ieee.org/document/8088163/

[13] A. Y. Javaid, W. Sun, V. K. Devabhaktuni, and M. Alam, "Cyber security threat analysis and modeling of an unmanned aerial vehicle system," in *2012 IEEE Conference on Technologies for Homeland Security (HST)*. IEEE, Nov. 2012, pp. 585–590. [Online]. Available: http://ieeexplore.ieee.org/document/6459914/

[14] E. Vattapparamban, I. Guvenc, A. I. Yurekli, K. Akkaya, and S. Uluagac, "Drones for smart cities: Issues in cybersecurity, privacy, and public safety," in *2016 International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, Sep. 2016, pp. 216–221. [Online]. Available: http://ieeexplore.ieee.org/document/7577060/

[15] M. Heiges, R. Bever, and K. Carnahan, "How to Safely Flight Test a UAV Subject to Cyber-Attacks," Georgia Tech Research Institute, Tech. Rep., 2015.

[16] Y. Son, H. Shin, D. Kim, Y. Park, J. Noh, K. Choi, J. Choi, and Y. Kim, "Rocking drones with intentional sound noise on gyroscopic sensors," in *24th USENIX Security Symposium (USENIX Security 15)*. Washington, D.C.: USENIX Association, 2015, pp. 881–896. [Online]. Available: https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/son

[17] *ISO27005:2011 Information technology – Security techniques – Information security risk management*, International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) Std.

[18] C. Schmittner, T. Gruber, P. Puschner, and E. Schoitsch, "Security Application of Failure Mode and Effect Analysis (FMEA)," in *Computer Safety, Reliability, and Security*, A. Bondavalli and F. Di Giandomenico, Eds. Cham: Springer International Publishing, 2014, vol. 8666, pp. 310–325. [Online]. Available: http://link.springer.com/10.1007/978-3-319-10506-2_21

[19] B. Schneier, "Modeling security threats," *Dr. Dobb's Journal*, 1999. [Online]. Available: https://www.schneier.com/academic/archives/1999/12/attack_trees.html

[20] I. Sabir, R. Yves, F. Michael, L. Timo, F. Andreas, G. Sigrid, H. Olaf, R. Roland, R. Matthias, B. Henrik, A. Ludovic, P. Renaud, P. Gabriel, R. Alastair, W. David, and W. Benjamin, "Security requirements for automotive on-board networks based on dark-side scenarios," EVITA, Tech. Rep., 2009.

[21] J. Xu, K. K. Venkatasubramanian, and V. Sfyrla, "A methodology for systematic attack trees generation for interoperable medical devices," in *2016 Annual IEEE Systems Conference (SysCon)*, April 2016, pp. 1–7.

[22] E. J. Byres, M. Franz, and D. Miller, "The use of attack trees in assessing vulnerabilities in SCADA systems," in *in IEEE Conf. International Infrastructure Survivability Workshop (IISW 04). Institute for Electrical and Electronics Engineers*, 2004.

[23] E. Kelling, M. Friedewald, T. Leimbach, M. Menzel, P. Sger, H. Seudi, and B. Weyl, "Specification and evaluation of e-security relevant use cases. deliverable d2.1: Evita. e-safety vehicle intrusion protected applications," 09 2007.

[24] *AIRWORTHINESS SECURITY PROCESS SPECIFICATION ED-202 / DO-326*, EUROCAE, Jun. 2014.

[25] *JURAS guideline on SORA*, Joint Authorities for Rulemaking on Unmanned Systems, 2017, annex A : Guideline on collecting and presenting system and operation information for a specific UAS operation.

[26] *AIRWORTHINESS SECURITY METHODS AND CONSIDERATIONS*, EUROCAE, 102 rue Etienne Dolet, 92240 MALAKOFF, France, Sep. 2015.

[27] O. Gadyatskaya, R. Jhawar, P. Kordy, K. Lounis, S. Mauw, and R. Trujillo-Rasua, "Attack trees for practical security assessment: Ranking of attack scenarios with adtool 2.0," in *Quantitative Evaluation of Systems*, G. Agha and B. Van Houdt, Eds. Cham: Springer International Publishing, 2016, pp. 159–162.