Sphere: a novel platform for increasing Safety & Security on Unmanned Systems

Daniel Fernando Pigatto
Institute of Mathematical and Computer Sciences, University of São Paulo
São Carlos, SP, Brazil
and
University of the West of England
Bristol, UK
Email: pigatto@icmc.usp.br

James Smith
University of the West of England
Bristol, UK
Email: James.Smith@uwe.ac.uk

Kalinka Regina Lucas
Jaquie Castelo Branco
Institute of Mathematical and Computer Sciences, University of São Paulo
São Carlos, SP, Brazil
Email: kalinka@icmc.usp.br

***Abstract*—The Healthy, Mobility and Security-based Data Communication Architecture, also known as HAMSTER, is provided with a special platform for safety & security: Sphere. It concentrates all the safety & security aspects of the main architecture and all derivative versions. The aim is to define patterns for assuring safety & security that allow every unmanned vehicle derived from HAMSTER to safely share information, even when different scenarios are involved, e. g. to permit the safe communication between an unmanned ground vehicle and an unmanned aerial vehicle. It is also a goal of Sphere to centralize the modules "health" check, which guarantees a safer operation for the vehicle and, consequently, the entire system. Modules and subsystems criticality measurements are proposed as part of the architecture definition for research &development of robust, safe, health and secure unmanned vehicles and systems.**

I. Introduction

The development of Unmanned Vehicles (UV) and Unmanned Systems[1] has increased recently, fact that allowed the existence of several different types of vehicles e. g., aerial, terrestrial and aquatic vehicles. Such vehicles are likely to be integrated into the airspace, on public roads and even on aquatic environments following specific laws and requirements of each scenario. Thus, it is essential that all modules and subsystems that compose the aircraft as full communications elements meet healthy, mobility and security requirements, increasing the system overall capabilities and therefore allowing the vehicles to be certified and integrated into their operation space, following the specific rules determined by authorities, which may vary from country to country.

This paper presents Sphere, the Safety & Security platform from HAMSTER architecture [1]. The main objective of such platform is to help Researchers & Developers of UV to efficiently implement safety & security in their systems in a very integrated way. The motivation for developing Sphere is that safety & security must not be considered as features to be posteriorly integrated to an architecture, vehicle or system. Contrarily, they must be developed accordingly with the other UV subsystems, providing full possibilities for increasing the overall system robustness, safety and the information security.

This paper is organized as follow: Section II presents an overview of HAMSTER architecture; Section III presents a review of what has been done in the literature and what is still an issue for safety & security in UVs; Section IV brings all the details of the Sphere platform; Section V provides a criticality classification proposal for modules and subsystems; and Section VI concludes the paper.

II. HAMSTER Architecture

The Healthy, Mobility and Security-based Data Communication Architecture was divided into three main versions according to the most common types of UV: aerial, aquatic and terrestrial. It was also defined two extra modules: one to deal with safety & security aspects under all three versions of HAMSTER, and another one responsible for all the mobility aspects in such systems. Figure 1 presents an overview of HAMSTER hierarchical organization.

Flying HAMSTER is the version of the architecture which deals exclusively with the aerial segment. It was defined based on specific characteristics and requirements of unmanned aerial vehicles (UAV) and unmanned aircraft systems (UAS). Flying HAMSTER deals specifically with internal airplane communication (IAC), airplane-to-airplane communication (A2A) and airplane-to-infrastructure communication (A2I).

The main applications of UAVs are related to agricultural and environmental monitoring, safety, military and civil defense. The aircraft is usually able to capture images for processing relevant information about a specific field, which may contribute to improve productivity. There are several cases where they might be applied in environmental and borders monitoring, or even applied as aerial sensors

[1]Unmanned Systems, in this paper, refers to everything present in a limited environment that allows the execution of a mission, e. g., the Unmanned Vehicle, the Ground Station etc.

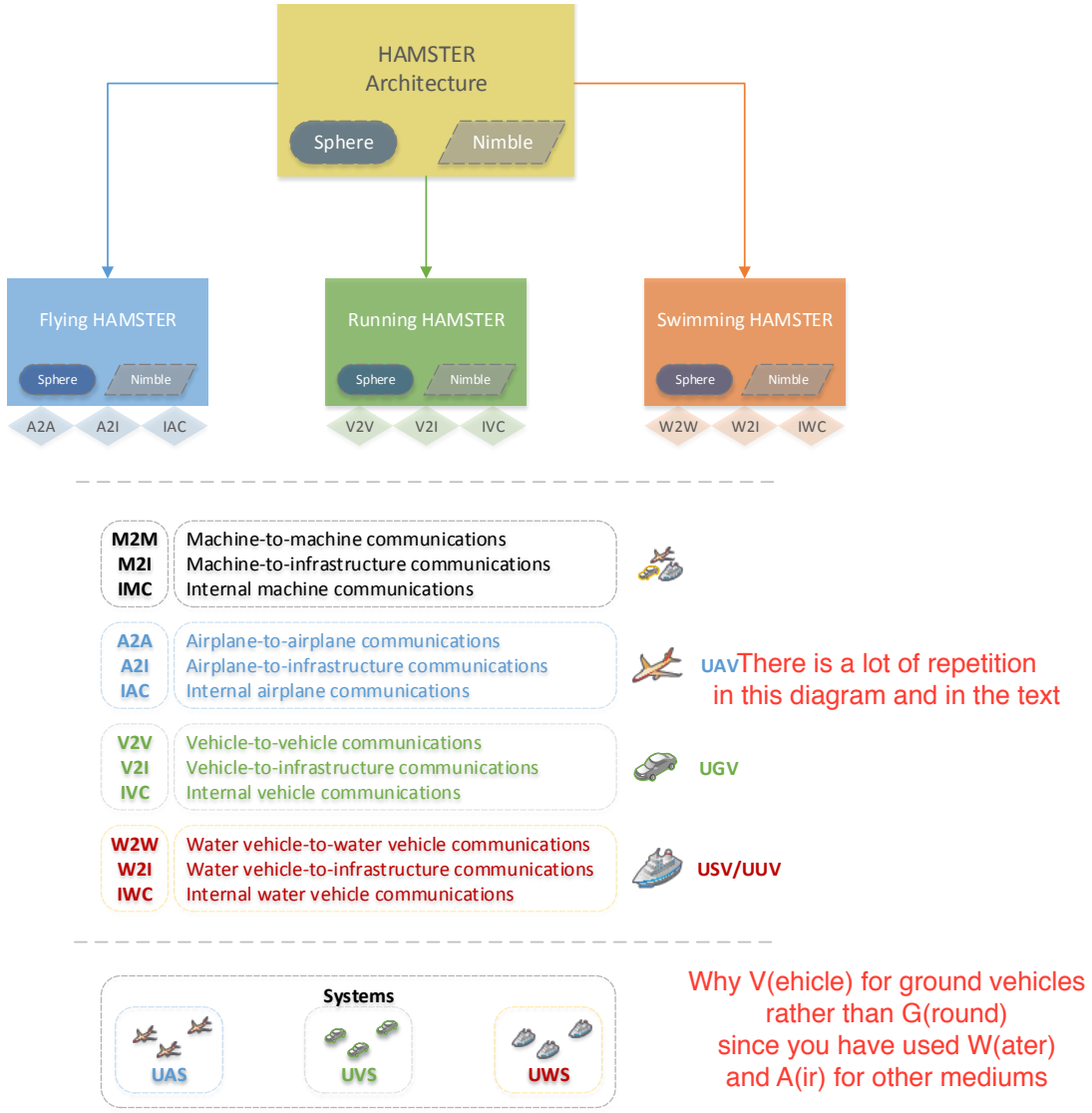

Figure 1. HAMSTER versions and the specific modules for mobility (Nimble) and safety & security (Sphere). Flying HAMSTER was designed for aerial systems (UAV and UGS), Running HAMSTER for terrestrial systems (UGV and UGS) and Swimming HAMSTER for aquatic systems (USV, UUV and UWS). Unmanned aircraft systems (UAS) are composed by unmanned aerial vehicles (UAV) and every other instance that communicates with the system to perform a mission through A2A, A2I and IAC communications; Unmanned vehicles systems (UVS) are composed by unmanned ground vehicles (UGV) and every other instance that communicates with the system to perform a mission through V2V, V2I and IVC communications; Unmanned water vehicles systems (UWS) are composed by unmanned surface/underwater vehicles (USV/UUV) and every other instance that communicates with the system to perform a mission through W2W, W2I and IWC communications. Every established communication is derived from the general concepts represented by M2M, M2I and IMC communications.

in networks for disaster management [2] and multiple UAV applications [3], [4], [5], [6], [7].

Running HAMSTER deals specifically with vehicles on terrestrial segment. It was defined based on specific characteristics and requirements of unmanned ground vehicles (UGV) and unmanned ground systems (UGS). Running HAMSTER treats internal vehicle communication (IVC), vehicle-to-vehicle communication (V2V) and vehicle-to-infrastructure communication (V2I).

The objective of ground vehicles may vary from driver support in possible dangerous situations with the intention of preventing road accidents, to autonomous driving with no human intervention, which could be used in urban traffic, agriculture, industry and safety applications [8]. The sensor fusion technique is used for integration of multiple sensors such as cameras, digital compasses, and GPS, allowing the vehicle to become autonomous in both urban and rural areas [9].

Swimming HAMSTER was designed for vehicles that operate on aquatic environments. It was defined based on specific characteristics and requirements of unmanned surface vehicles (UGV), unmanned undersea vehicles (UUV) and unmanned water vehicles systems (UWS). Swimming HAMSTER is composed by internal water vehicle communication (IWC), water vehicle-to-water vehicle communication (W2W) and water vehicle-to-infrastructure commu-

nication (W2I).

The aquatic vehicles have been used for various tasks, especially those related to monitoring of oil exploration and maintenance of hydropower. The current challenges for these vehicles go beyond autonomy, integrating other areas with the distributed and embedded systems, such as computer networks, artificial intelligence, software engineering, electrical, mechanical and mechatronics engineering, among others. The multiple vehicles tasks are also challenging [10], [11].

All versions of HAMSTER possess modules that compose the overall safety & security platform, called Sphere.

III. Safety & Security Issues for UV

Chien and Lin [12] made a study of a protocol for key exchange in a mobile ad hoc wireless network (MANET). It was stipulated that the network would be composed of three layers. A military example was used, in which the first layer is composed of nodes that communicate with a central unit within the backbone network. In the example, the nodes are soldiers sending information and the backbone is a vehicle with more computational power. The second level consists of several backbones composing a wireless network. Finally, the third level is a UAV flying in the area of the backbone, making the centralized network.

Eissa *et al.* [13] focused on creating an authentication mechanism in wireless ad hoc networks. To do this, four different keys are generated. An identity key, a public key, a private key, and symmetric key. The work assumes communication between AN and BN nodes. The first step is to check the confidence level of both nodes, which is done by querying the neighboring nodes using the identity key. If at least n nodes confirm the level of confidence of a specific node, the communication begins. The nodes AN and BN agree on a session key using the public key and keep it in their trusted keys database. Thereafter, nodes use their private key to encrypt messages. As a result, cryptanalysis attacks do not work because the public key is required. The Sphere approach, which will be presented in Section IV, considers different methods of verifying the nodes authenticity.

Faughnan *et al.* [14] made a study of UAVs kidnapping. The method is divided into two parts. The first one is the risk identification of an attack on the UAV. To perform this part, a list of risk scenarios was created. The second one consists on the creation of a mechanism to inform the system operator that the UAV is under attack. There are two systems onboard the plane measuring system speed. If there is great variation in the speed measurement, it might indicate that the system is under attack. For testing it was used a moving car to simulate the UAV. The result was a framework capable of detecting attacks using the measure of an essential variable. Such risk identification could be integrated in a central unit provided by Sphere, which will be better explained in Section IV.

Kashikar and Nimbhorkar [15] presented the exchange of messages among nodes in a MANET. As this exchange of messages is done by exchanging a data packet at a time, such networks are affected by DoS attacks. The proposed method aims to block access to the network by malicious nodes. If the target node of the attack begin receiving packages in quantities larger than the network supports, this node will prompt the attacker node to decrease its transmission rates. If not, the communication between the nodes is stopped and the target node will put the attacker node in a list of unreliable identifications. When a node attempts to join the network, the nodes that had already joined make a check on their lists of unreliable identifications. If the identification of the node which is trying to join the network is in one of such lists, it will not be allowed to enter the network.

Bakar *et al.* [16] aimed to study the creation of secure channels for communication between UAV systems, satellites and base stations. The paper addresses the major attacks in UAVs. Initially, it were determined the main components of the system, based on the degree of criticality. Then, it was created a system model and associated attacks and threats. The results were tested in a simulator in order to analyze the behavior of the network under attack. After a series of attacks, the system had some failed components, especially after the denial of service attack.

Man *et al.* [17] made a study monitoring the health and safety of UAV systems. As a basis for study, it was designated a model with the main components of an aircraft. These components have been grouped according to their function in the UAV. Thus, in case of errors in a module, the path of propagation of this error is known. In addition, the author discussed some techniques to predict when the modules should begin to be defective based on the quality of data and experience regarding the use of UAVs. The paper addresses important concepts in the health of the UAV. It was not taken into account the criticality of each module, which is part of the Sphere proposal (refer to Section IV for more details).

Raj *et al.* [18] studies a protocol for the admission of nodes on a network in a decentralized manner. In the beginning of the network it is determined that there are only trusted nodes. This group of nodes has a shared secret key. To enter the network, a node must make the request and receive permission from all nodes in the network using a secure channel of communication. If the node is approved for communication, network nodes create a new shared secret key and all nodes exchange keys for use in pairs of nodes at the time of secure communication.

Yedavalli and Belapurkar [19] presented applications of wireless sensor networks in aircraft, including the major systems of the aircraft and system limitations. In wireless networks, for example, it is possible to distribute the aircraft engine control, exchanging common sensors for smart ones. Another advantage is the reduction of the aircraft system weight, which leads to the vehicle being able to carry more loads. The paper also discusses some challenges of this approach, such as restrictions on the control of communication and best way to provide power to the sensors.

IV. Sphere: Safety & Security Platform on HAMSTER Architecture

Sphere is the platform for safety & security in HAMSTER architecture. Every aspects related to such subjects will be represented inside Sphere. Although the platform may have centralized modules, it is not a centralized platform. Sphere is present in many parts of the UV according to its necessities. It is responsible from information security (the way it is exchanged, stored, manipulated etc.) to healthy and safety of the overall UV and all subsystems that compose unmanned system.

A. Why Sphere?

The name Sphere comes from the idea of a hamster ball that allows the animal to play in a safe way. As an sphere has the visual concept of wrapping things, it was chosen as the name of the safety & security platform for HAMSTER architecture. It concentrates all the safety & security aspects of the main architecture and all derivative versions. The aim is to define patterns for assuring safety & security that allow every unmanned vehicle derived from HAMSTER to safely share information, even when different scenarios are involved, e. g. to permit the safe communication between an unmanned surface vehicle and an unmanned aerial vehicle. It is also a goal of Sphere to centralize the modules "health" check, which guarantees a safer operation for the vehicle and, consequently, the entire system.

Section IV-B will present details about how Sphere works.

B. The Sphere Main Proposal

This subsection is divided into two parts. The first one addresses a components usage policy and the second one an authentication protocol which will also be responsible by the components "health" checking.

1) Components usage policy: One of the first steps to ensure the safe operation of a vehicle and to facilitate its integration into the space of actuation (for instance, an UAV into the airspace) must be the redefinition of its components usage policy. Only a few parts of an UV are properly treated to ensure that all connected modules are authentic and have not been replaced or tampered with by a third party. The current policy adopted by most aircraft manufacturers uses a concept of "Accept all" which trusts in all components embedded in an aircraft. This proposal suggests the assumption of an "Almost Deny All" approach, which denies the authenticity of all mechanical components and peripherals attached to the vehicle until the opposite is proved, which may result in safer vehicles.

The categorization of every module is therefore crucial for such a new security model to be applied to UV. There are various peripheral devices embedded in an UV that require different levels of security, which leads to the necessity of a module categorization according to the criticality of their performed functions. The Sphere proposal suggests the modules categorization into primary, secondary, and so on, according to necessity. Bigger and more robust vehicles may have their modules divided into more than two categories, once there is a greater variety of modules criticality to be considered.

Primary modules are those considered essential components for the UV to operate, to be aware of its location and to be able to perform an emergency operation abort safely, even when the mission was not entirely concluded. An autopilot, a GPS receiver, and barometric/inertial units are examples of modules classified as primary, since they might cause a big lost if in failure state. In contrast, modules not considered as essential functions to the UV are classified as secondary modules. Whether abnormal behaviors are detected in any secondary module, the operation of the primary components of the UV is not affected and the secondary module that presented the abnormality should be disabled or isolated. It implies that all primary modules must be authenticated before the operation begins. However, the secondary modules do not necessarily need an authentication before the mission execution.

In addition, to protect the UV against malicious attacks, there is the possibility of identifying anomalies due to usage time. For instance, pressure and collisions suffered by an aircraft may cause natural degradations in components integrity. Therefore, mechanisms to identify the existence of unusual behaviors should help to increase the UV safety, even with a consequent abort of a mission for reasons of physical integrity of the UV. These concepts are strongly connected to Sense & Avoidance area, which are considered as a feature to be integrated to HAMSTER as a future work.

The creation of access profiles for modules is another concept associated with the proposal of authentication. As a mission is assigned to the UV, it must go through an authentication process, which assigns different access permissions to the UV modules. Such concept is similar to the one used in modern operating systems where an administrator user is allowed to install and uninstall software with no restrictions, unlike a visitor user who has access to the programs, but is not allowed to install/uninstall them. Applied to UV, such concept adds a layer of security that allows blocking the use of selected modules by specific users. Such specification is intended to prevent unauthorized access. Even if there is a single effective user of an UV, no other user (an attacker or not) will have privileged access to modules or their information.

Figure 2 presents the Sphere main modules. Central Security Unit (CSU) is responsible by modules authentication and "health" verification. Based on such results, CSU associates every module to a particular usage profile. The communication security is also addressed by Sphere and directly impacts the respective vehicle communications. The access policy assigned to each user must use cryptographic algorithms suitable for embedded or real-time sensitive environments. Several experiments have been carried out regarding security for critical embedded systems [20], [21].

2) Protocol structure: To protect the UV against attacks coming from malicious components, Sphere implements strict security policies. It is necessary to ensure that all modules are authentic, so if one of them fails

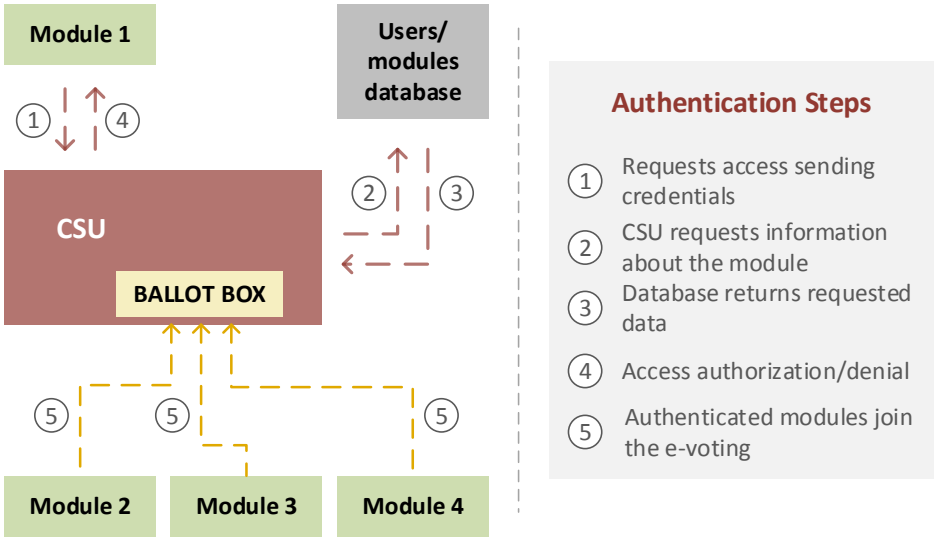

Figure 3. Modules and CSU authentication steps. The "Ballot box" is used by every other system modules for authenticating CSU through e-voting protocols.

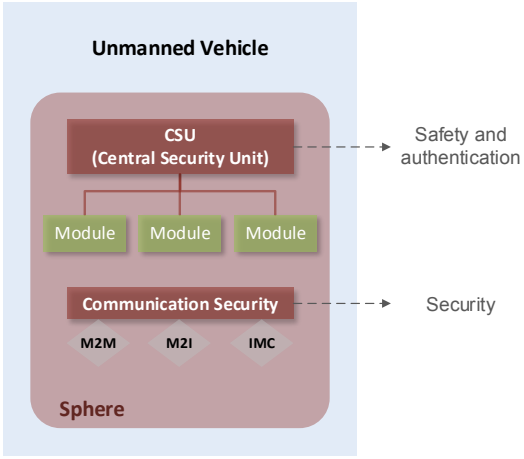

Figure 2. Generic composition of Sphere, the Safety & Security aspects module on HAMSTER architecture. Each version of HAMSTER may address several different concepts, according to specific necessities. However, CSU will be able to equally communicate among different HAMSTER versions, allowing heterogeneous vehicles communications.

or presents an abnormal behavior, the others must not communicate with it. Furthermore, such policies must be applicable even during vehicle operations, considering that external factors may affect the components behavior e. g. climate or weather changes. In addition, each component must contribute for overall UV safety. In order to apply such methods and requirements, it is assumed that at the system startup or after hardware changes, CSU module remains in an unsafe state and must be authenticated. It will be responsible for storing a table of public keys of all vehicle components, operating similarly to a Certification Authority (CA), which has the goal of ensure that a public key belongs to an entity (module). Each module (or component) will store a hash table of the keys for integrity checking.

During the vehicle start up, a mutual authentication phase should occur with CSU. It checks the database credentials of all modules, their criticality, and even if there is any access restriction. There is also the possibility of deciding whether a module should be initialized or not during the verification stage. The following steps will be authentication and exchange of encrypted messages to establish a secure channel for communication among modules and CSU.

After such handshake, three situations are expected:

- The module that is trying to authenticate and CSU have not been tampered with;
- The module has been tampered with and therefore has not been authenticated:
  - If it is a module of primary type, the UV must not operate;
  - If it is a module of secondary type, communication with it must be interrupted and a notification be sent to a control station.
- The module that is trying to authenticate may notice that CSU is not authentic, and must notify other components about it.

From the point of view of communication security, an ideal situation would be if all modules could authenticate with others. However, this method would cause a system overload, since the increase of modules in the aircraft would cause an exponential increase in the number of exchanged messages. To solve such problem there exist the e-voting protocols [22]. In case of non-authentic CSU, protocols such as those presented in [23] might be used. Such model can be further expanded according to the needs of the UV, including a negotiation mediated by CSU to create

a secure channel of communication among modules. A graphical representation of processes performed during the authentication module with CSU can be seen in Figure 3.

C. How is Sphere integrated to the HAMSTER architecture?

The CSU proposal is aimed at authenticating every module and keeps important information about all of them. Such database owned by CSU is capable of providing more than just authenticity credentials about modules or subsystems. Furthermore, it may contain criticality information and own routing tables for establishing secure communications among modules and subsystems of UVs.

Such integration makes CSU not just the central unit for safety & security information, but also the coordinator of communications, permitting it to allow/deny specific communications, in cases it is needed. Profiles and subsystems criticality profiles are current results of Sphere proposal that will be presented in Section V.

V. Results and Discussion

The very first step of an architecture definition must be the identification of criticality in general modules that compose an UV and a ground station, allowing appropriate approaches for each of such elements, ensuring the proper operation of the UV. Such criticality classification will be defined by a set of characteristics that may vary from system to system, but a general analysis will help defining strategies for real time systems regarding communications, safety & security.

Our proposal of modules and subsystems criticality classification is composed by several information. Considering an example of the Air Data Sensors subsystem, which is composed by dynamic pressure, static pressure, rate of change of pressure and temperature. The aim is to provide data such as barometric altitude, indicated airspeed, vertical speed, Mach, static air temperature, total air temperature and true airspeed [24].

Sphere determines that a 'form' with Air Data sensor subsystem characteristics must be filled providing important information for determining its criticality in the overall system. Such information follow the rules that can be seen in Table I. Every sensor that compose the Air Data Sensor subsystem of our example (pressure sensor, temperature sensor, air speed sensor and altitude sensor) will also have a specific form following the rules for modules profiles, which can be seen in Table II.

Based on such classification presented in Tables I and II it will be possible to compose the subsystem/module profile in a specific system. Extra information may contribute for classification e. g. if the area where the mission is being performed is populated, the required altitude for performing a mission, and the existence of obstacles in the area, to name a few.

Regarding the criticality level, four situations are expected:

- High Criticality:
  1) *Catastrophic*: when a failure may cause the loss of the UV.
  2) *Critical*: when a failure may cause the loss of almost the entire mission.
- Low Criticality:
  3) *Marginal*: when a failure may compromise a small part of the mission.
  4) *Minor*: when a failure would not cause problems neither to the aircraft or the mission, but it would require a posterior repair to the subsystem/module.

Sphere will consider the criticality classification for defining priority of communication, safety & security in the overall system. Such results are the start of the Sphere proposal creation and will be implemented very close to the HAMSTER architecture implementation.

Table I. Subsystem profile form with the key information for composing the profile of the vehicle subsystems.

Subsystem Profile	
Name	Name of the subsystem being described.
Modules	Names of the modules that compose the subsystem.
Function	Brief description of subsystem function for the system.
Aircraft size	1. General aircraft 2. Small 3. Medium 4. Big
Criticality level	High Criticality: 1. Catastrophic 2. Critical Low Criticality: 3. Marginal 4. Minor
Criticality phases	1. Take-off 2. Landing 3. Flying 4. All the time
Reasons	According to the type of aircraft and missions performed, reasons about the chosen criticality may vary.

Table II. Module profile form with the key information for composing the profile of the vehicle modules.

Module Profile	
Name	Name of the module being described.
Subsystem	Names of the subsystem it belongs to.
Function	Brief description of subsystem function for the system.
Aircraft size	1. General aircraft 2. Small 3. Medium 4. Big
Criticality level	High Criticality: 1. Catastrophic 2. Critical Low Criticality: 3. Marginal 4. Minor
Criticality phases	1. Take-off 2. Landing 3. Flying 4. All the time
Reasons	According to the type of aircraft and missions performed, reasons about the chosen criticality may vary.

VI. Conclusions

Safety & security compose an important paradigm for every critical embedded system. The development of systems that consider such paradigm since the beginning of its development are more capable of providing safety & security in a better way regarding performance. Unmanned vehicles developed following the guidelines of HAMSTER architecture will find a platform that completely support the development of such paradigm in every stage of the process. It is also prepared for covering components "health" check, overall safety improvements and information security guarantee in every exchange, storage or manipulation of data.

Sphere is under the HAMSTER architecture to provide not only increased safety & security for one vehicle, but also for groups of vehicles either similar or not. Sphere was made to allow different HAMSTER based architectures to communicate. Thus, ground, aerial and aquatic vehicles are able to securely exchange important information, contributing for the interaction of different scenarios where unmanned vehicles may be applied.

This paper presented our proposal for increasing safety & security in unmanned vehicles. It also presented some preliminary results with criticality level definitions and the application of the concept of criticality profiles to classify modules and subsystems. Sphere will is in constant development under the HAMSTER architecture.

Acknowledgments

The authors would like to thank FAPESP for the financial support through processes 2012/16171-6 and 2014/13713-8.

References

[1] D. F. Pigatto, L. Goncalves, A. S. R. Pinto, G. F. Roberto, J. Fernando Rodrigues Filho, and K. R. L. J. C. Branco, "HAMSTER - Healthy, mobility and security-based data communication architecture for Unmanned Aircraft Systems," in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, May 2014, pp. 52–63. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6842238

[2] M. Quaritsch, K. Kruggl, D. Wischounig-Strucl, S. Bhattacharya, M. Shah, and B. Rinner, "Networked UAVs as aerial sensor network for disaster management applications," *e & i Elektrotechnik und Informationstechnik*, vol. 127, no. 3, pp. 56–63, Mar. 2010. [Online]. Available: http://link.springer.com/10.1007/s00502-010-0717-2

[3] I. Maza, F. Caballero, J. Capitán, J. Martínez-De-Dios, and A. b. Ollero, "Experimental results in multi-UAV coordination for disaster management and civil security applications," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 61, no. 1-4, pp. 563–585, 2011. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-79951517007&partnerID=40&md5=16bb6815a7cb9ff76f1974045691b41d

[4] O. Bouachir, A. Abrassart, F. Garcia, and N. Larrieu, "A mobility model for UAV ad hoc network," in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, May 2014, pp. 383–388. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6842277

[5] C. Luo, P. Ward, S. Cameron, G. Parr, and S. McClean, "Communication provision for a team of remotely searching UAVs: A mobile relay approach," in *Globecom Workshops (GC Wkshps), 2012 IEEE*, 2012, pp. 1544–1549.

[6] S.-W. Kim and S.-W. Seo, "Cooperative Unmanned Autonomous Vehicle Control for Spatially Secure Group Communications," *Selected Areas in Communications, IEEE Journal on*, vol. 30, no. 5, pp. 870–882, 2012.

[7] A. Verma and R. Fernandes, "Persistent unmanned airborne network support for cooperative sensors," in *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 8756, Baltimore, MD, 2013. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84881189069&partnerID=40&md5=ffb4981b4b75aa0fafdbc8017bab60ce

[8] L. C. Fernandes, J. R. Souza, G. Pessin, P. Y. Shinzato, D. Sales, C. Mendes, M. Prado, R. Klaser, A. C. Magalhães, A. Hata, D. Pigatto, K. Castelo Branco, V. Grassi, F. S. Osorio, and D. F. Wolf, "CaRINA Intelligent Robotic Car: Architectural design and applications," *Journal of Systems Architecture*, vol. 60, no. 4, pp. 372–392, Apr. 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1383762113002841

[9] Z. Sun, P. Wang, M. Vuran, M. Al-Rodhaan, A. Al-Dhelaan, and I. b. Akyildiz, "BorderSense: Border patrol through advanced wireless sensor networks," *Ad Hoc Networks*, vol. 9, no. 3, pp. 468–477, 2011. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-78651354776&partnerID=40&md5=d95c68ba2dbbd4b9d4805179711b1d0

[10] X. Xiang, C. Liu, L. Lapierre, and B. Jouvencel, "Synchronized path following control of multiple homogenous underactuated AUVs," *Journal of Systems Science and Complexity*, vol. 25, no. 1, pp. 71–89, 2012. [Online]. Available: http://link.springer.com/10.1007/s11424-012-0109-2

[11] S. Abbott-Mccune, P. Kobezak, J. Tront, R. Marchany, and A. Wicks, "UGV: Security analysis of subsystem control network," in *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 8741, Baltimore, MD, 2013. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84881179395&partnerID=40&md5=44ebd9855c951ede2a29c3d96676efe9

[12] H.-Y. Chien and R.-Y. Lin, "Identity-based key agreement protocol for mobile ad-hoc networks using bilinear pairing," in *Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006. IEEE International Conference on*, vol. 1, 2006, pp. 8 pp.–.

[13] T. Eissa, S. A. Razal, and M. A. Ngadi, "Enhancing MANET Security Using Secret Public Keys," in *Future Networks, 2009 International Conference on*, 2009, pp. 130–134.

[14] M. S. Faughnan, B. J. Hourican, G. C. MacDonald, M. Srivastava, J. A. Wright, Y. Y. Haimes, E. Andrijcic, Z. Guo, and J. C. White, "Risk analysis of Unmanned Aerial Vehicle hijacking and methods of its detection," in *Systems and Information Engineering Design Symposium (SIEDS), 2013 IEEE*, 2013, pp. 145–150.

[15] M. Kashikar and S. Nimbhorkar, "Designing acknowledgement based MANET using public key cryptography," in *Computer Science Education (ICCSE), 2013 8th International Conference on*, 2013, pp. 228–233.

[16] A. A. Bakar, R. Ismail, A. R. Ahmad, J.-L. Abdul, and J. Jais, "Group based access control scheme (GBAC): Keeping information sharing secure in mobile Ad- hoc environment," in *Digital Information Management, 2009. ICDIM 2009. Fourth International Conference on*, 2009, pp. 1–6.

[17] Q. Man, S. Ma, L. Xia, and Y. Wang, "Research on security monitoring and health management system of medium-range UAV," in *Reliability, Maintainability and Safety, 2009. ICRMS 2009. 8th International Conference on*, 2009, pp. 854–857.

[18] E. Raj, S. SelvaKumar, and J. R. Lekha, "Node admission protocols for secure communications," in *Emerging Trends in Electrical and Computer Technology (ICETECT), 2011 International Conference on*, 2011, pp. 69–73.

[19] R. K. Yedavalli and R. K. Belapurkar, "Application of wireless sensor networks to aircraft control and health management systems," *Journal of Control Theory and Applications*, vol. 9, no. 1, pp. 28–33, Mar. 2011. [Online]. Available: http://link.springer.com/10.1007/s11768-011-0242-9

[20] V. Schoaba, F. E. G. Sikansi, D. F. Pigatto, K. R. L. J. C. Branco, and L. C. Branco, "Digital Signature for Mobile Devices: A New Implementation and Evaluation," *International Journal of Future Generation Communication and Networking*, vol. 4, pp. 23–36, 2011.

[21] D. F. Pigatto, N. B. F. D. Silva, and K. R. L. J. C. Branco, "Performance Evaluation and Comparison of Algorithms for Elliptic Curve Cryptography with El-Gamal based on MIRACL and RELIC Libraries," *Journal of Applied Computing Research*, vol. 1, no. 2, pp. 95–103, Feb. 2012. [Online]. Available: http://www.unisinos.br/revistas/index.php/jacr/article/view/1789

[22] H.-T. Liaw, "A secure electronic voting protocol for general elections," *Computers & Security*, vol. 23, no. 2, pp. 107–119, 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404804000276

[23] H. Kikuchi and J. Nakazato, "Modint: A Compact Modular Arithmetic Java Class Library for Cellular Phones, and its Application to Secure Electronic Voting," in *Security and Protection in Information Processing Systems*, 2004, pp. 177–192.

[24] I. Moir, A. Seabridge, and M. Jukes, *Civil Avionics Systems*, ser. Aerospace Series. Wiley, 2013. [Online]. Available: https://books.google.co.uk/books?id=8XFwAAAAQBAJ