Towards Embedded Intrusion Detection in Satellites

Louis Lolive
LAAS-CNRS / IRT Saint-Exupery
Toulouse, France
louis.lolive@laas.fr

Guillaume Auriol
LAAS-CNRS / INSA Toulouse
Toulouse, France
guillaume.auriol@laas.fr

Vincent Nicomette
LAAS-CNRS / INSA Toulouse
Toulouse, France
vincent.nicomette@laas.fr

***Abstract*—Satellites are widely used in our day-to-day life, which makes them a strategic target. Thus, the security of these satellites must be seriously considered, as showcased by recent attacks. A security by-design approach is of course essential, but complementary security measures are also needed, such as Intrusion Detection Systems (IDSs). In this paper, we describe the design of an intrusion detection system embedded in the satellite, as well as the corresponding threat model. The IDS, still in development, uses a multi-probe and multi-modules approach, to be able to face multiple threats. The paper also briefly describes some attack scenarios that we are currently implementing to assess the relevance of the IDS. The experiments are carried out on the NASA Operational Simulator for Small Satellites, the NOS^3 platform.**

***Index Terms*—security, space, embedded, detection, cyber, cubesat, attack**

I. Introduction

Throughout the years, space systems have become more and more present in our daily lives. Satellites networks are essential to our day-to-day communication and travels. They also play a strategic role in national defense (observation satellites, for example). As a consequence, and taking into account the increasing number of cyberattacks around the world, satellites systems have become strategic targets. For a long time, space systems have relied on their complexity, opacity and relative unreachability to protect themselves from cyberattacks (with state means being needed to impact them), but recent examples, such as the VIASAT attack [1], the OPS-SAT hack during the CYSAT conference [2], or the successful takeover in the 4th edition of Hack-A-Sat of a CubeSat named Moonlighter [3], have shown that the security of satellites must be seriously considered. Furthermore, the quick growth of the New Space area, and the increasing use of Commercial-Off-The-Shelf (COTS) components in low-cost and small-scale satellites (e.g. CubeSats, NanoSats...) means that aforementioned security by obscurity is starting to become less and less obscure to attackers. Therefore, current research is heading towards new ways of improving the security of space systems, such as security by-design [4] and intrusion detection systems.

The security of space systems must be considered at different strategic locations: on-ground systems, communications links and embedded systems (on the satellite itself). While many security mechanisms to protect on-ground information systems exist, less research works aiming at embedding security mechanisms inside the satellite itself are published as for now. This is the purpose of our research work that focuses on Intrusion Detection Systems (IDS) embedded in satellites. This work is challenging because of the specific characteristics of the satellites: the resources available are limited, the system must be autonomous in the sense that 1) no human being can intervene on board of the satellite and 2) a satellite is, from time to time, not in visibility of the ground station. Some of these characteristics are common to other embbeded systems (e.g. airplaines, automotive systems), while others, such as the autonomy are more specific. The IDS described in this paper is based on a multi-probe and multi-modules approach. This architecture has been designed so that a large variety of attack scenarios are covered by the detection. The IDS is currently implemented on the NOS3 plateform from NASA.

This paper is organized as follows. The Section II briefly describes related research work, Section III gives context on the simulator and the considered satellite, Section IV presents the intrusion detection system that we currently design and implement, Section V describes attacks developed with the aim of evaluating and training the IDS, and Section VI will summarize current work and perspectives.

II. Related Work

Few examples of Intrusion Detection Systems embedded in satellites are published. In [5], main principles of what should be an IDS for CubeSats are given, but no implementation or experiments are available. In [6], under the assumption that the satellite is equipped with a CAN bus, an IDS using machine learning detection modules is proposed to detect attacks. A combination of time-based (i.e. timestamps) and content-based (i.e. payload) modules is used to improve efficiency, with content-based modules being moved to the ground segment to keep resource consumption low in the satellite. High accuracy, precision and recall are achieved for most of the attacks considered. In [4], an analysis of the architecture of the NOS^3 simulator [7] and its vulnerabilities is performed. System architecture solutions are discussed, including embedded detection systems, but no specific details are provided. In [8], a multi-probes and multi-solution architecture for an IDS embedded in satellites is recommended. The importance of explainable IDSs is stressed, even more so when using Machine Learning.

This work was supported by IRT Saint-Exupéry in the scope of the CSS project.

Other IDSs dedicated to space systems are located on the ground segment, and perform packet inspection [9] [10]. In [9], an hybrid IDS architecture combining signature-based detection and anomaly-based detection is proposed. In [10], four hybrid intrusion detection algorithms using Machine Learning and Deep Learning are proposed and compared. This on-ground detection is of course relevant, as satellites are mostly communicating with the ground, but also has drawbacks. Indeed, the ground segment location only allows to perform intrusion detection on data forwarded over-the-air by the satellite, which implies limited context (or quantity of data). As such, these intrusion detection mechanisms may not detect some on-board malware, especially when the satellite is not able to communicate with the ground. Furthermore, most of these approaches focus on security through the spectrum of satellite-terrestrial networks and Software-Defined Networking (SDN), dedicated to network data for detection, whereas we consider that system data could also be used to perform on-board detection.

If we extend the study of related work to other embedded contexts, such as automotive, IoT, etc, two main detection strategies seem to emerge. On one hand, Machine Learning, and especially Deep Learning, is used to learn a normal behavior and distinguish abnormal activities. On the other hand, non-AI techniques are used to provide explainable detection for specific systems.

Deep Learning [11] [12] [13], is used to learn the characteristics of communication data, whether on buses, or interfaces (for IoT). These approaches provide in many cases interesting results but are highly dependent on the quality and quantity of training data. As in our case, no datasets for satellites are up to now available, these approaches are difficult to implement. In addition, Deep Learning detection models are working as black-box algorithms, and thus lack explainability. Last but not least, these algorithms in many cases require a lot of computing and memory resources, which may be not guaranteed in satellite context.

In [14] [15] [16], non-AI approaches are used. They include Finite-State Automata [14], Transition Matrix on ID sequences [15], or histogram-based detection [16]. These approaches have the benefit of being highly explainable, but need to be designed specifically for the target system with a deep understanding of its operation.

In summary, though promising, these approaches fail to give experiments or results about embedded satellite-specific IDS, or when given, the communication bus used is not space-specific. In this work, we choose to use NOS^3's Software Bus architecture, as it is a space-specific, widely-used bus designed for its genericity and re-usability. Furthermore, as much as Machine Learning approaches are able to deliver good results for detection tasks, other approaches will be preferred if possible because of the lack of data and for better explainability.

III. Simulator Context

The simulator used in this work is NASA's NOS^3. It is composed of three main components : a ground station simulator (COSMOS tool); a visualisation component, also responsible for feeding information to the sensors and calculating position (named 42); and the satellite simulator itself, based on the NASA core Flight Software (cFS).

The version of NOS^3 used is a custom fork based on v1.6.2, developed in the scope of the "Cyber Space Simulation"(CSS) Project led by IRT Saint-Exupéry.

In this version, the communications protocol used between the ground station and the satellite is the Consultative Committee for Space Data Systems (CCSDS) protocol. The Transfer Frame (TF) stack is implemented allowing to use TeleCommands (TC) to send commands to the satellite, and to receive TeleMeasures (TM), as in the original NOS^3 the TF implementation is incomplete.

The satellite simulator's behavior is as follows. Upon receipt of a message from the ground, the Command Ingest (CI) component forwards it to the Software Bus (SB), which interconnects the components of the satellite. The message is then received by all the components that subscribed to it. Involved components can then execute actions or return information via TM. Communications on the SB use CCSDS Space Packet Protocol (SPP).

IV. Intrusion Detection

This subsection describes the threat model that we consider and the main detection strategies that we currently investigate for the embedded IDS.

A. Threat Model

Except for inter satellites communications, satellites mostly communicate with the ground, which means that to target the satellite network, compromising the ground segment is the easiest for an attacker. A lot of research papers focus on this ground segment as an attack vector, such as [17], in which the vector considered is a compromised ground station sending illegitimate commands to the satellite.

Another strategy for attackers would be using a "rogue" antenna to send commands to the satellite, but accurate tracking of the satellite trajectory would be needed, with a low visibility time frame. In addition, if the satellite and the ground segment use cryptography, sending commands impersonating the ground becomes a hard task. Therefore, using the legitimate ground station to access the satellite system seems the easiest for an attacker.

Another interesting threat consists in compromising the software before being embedded in the satellite (especially the client payload), through some vulnerabilities in the supply chain. This is a common vector in IT security, and as many suppliers are needed to engineer a satellite, the risk increases. Furthermore, as updates for satellite software are performed during operation (especially for the payload), software could be compromised during operation. Under this threat model, the compromised software is running inside the satellite itself and

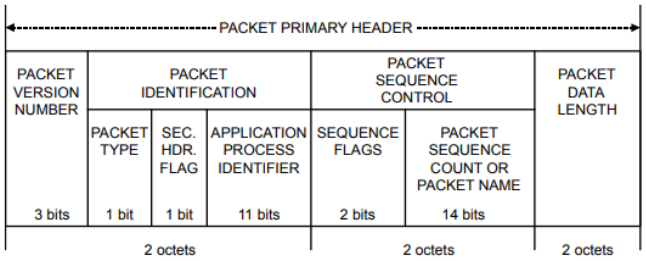

Fig. 1. Structure of a SPP packet primary header

cannot necessarily be detected by means of an IDS analyzing the communications between the satellite and the ground. As such, this leads to the design of an IDS embedded in the satellite itself. We assume that the whole flight system software is trusted, because we want to be able to trust state variables about the satellite supplied by the flight software. However, the payload software embedded in the satellite may be malicious and trigger malicious actions at any moment during the lifetime of the satellite.

Overall, in our research work, these two threats (compromission of the ground segment and compromission of the payload) are considered.

B. Attacks Scenarios

We assume that an attacker can take control of a ground station and craft as many TeleCommands (TC, i.e. commands sent to the satellite) as wanted. Invalid TC (with invalid SPP header identifiers) or valid TC with purposely crafted parameters can be sent. A representation of the primary header of a SPP packet is provided in Fig.1, in which identifiers are specified (secondary header is optional).

Two kinds of attacks have been considered.

- Some attacks may be performed simply by using one single TC which can have catastrophic consequences on the satellite. An example of such scenario is the sending a TC to turn off the calculators. This TC can be legitimate when sent to reboot a calculator, but could be used by an attacker to harm the satellite mission. Another example is the sending of a TC to put the satellite in a fail-safe configuration during the mission at an inadequate moment.
- Other more complex attacks may be performed by means of a set of apparently benign TC, but that, taken together, provoke the execution of malicious operations on the satellite. An example of these attacks is the sending of a set of TC to surreptitiously deviate the satellite from its correct trajectory. Another example is a set of TC allowing to upload a malicious file to replace a component of the satellite.

We assume that the payload has been compromised, either before launch or through an over-the-air update. The malicious actions that the payload can perform may target the integrity of the satellite itself, by trying to corrupt some flight system components for instance, but may also be more complex, for instance leaking some confidential information towards an accomplice on the ground, or purposely modifying the data managed by the payload itself to be sent to the ground. An example dealing with an observation satellite could consist in modifying the pictures taken by the satellite. The satellite could send fake photos, or purposely delete some specific parts of them.

C. IDS Overview

The IDS that we currently design is aimed to address these different attack scenarios. It uses a multi-layer approach with a set of probes (in different locations of the architecture), and a set of detection modules working either independently or together.

We investigate the design of three specific probes. The first, located at the arrival of each TC in the satellite (i.e. in the CI component of NOS^3), is aimed at capturing some features of the network traffic exchanged between the satellite and the ground segment: bandwidth, inter-arrival rate of packets, size of packets, etc. As we assume that cryptography is enabled, this probe is not able to get the content of the packet and thus can only get some characteristics at the network flow level.

The second probe is aimed at reporting information on deciphered TCs coming from the ground, and not sent yet on the Software Bus, to distinguish from internal messages. This probe is located into the CI component on the NOS^3 platform. This probe captures both the command type and command parameters and can provide them to an IDS dedicated to detect any malicious TC, as described above in the threat model.

The third probe is aimed at capturing information in the same way as the second one, but is located on the Software Bus. This positioning allows to gather information related to the internal exchanges of the satellite. The aim is to collect activity coming from all components of the satellite. This probe will be implemented either by modifying the software bus itself or by developing a specific application that subscribes to all messages on the bus (messages between the different components of the flight software, messages between the payload and the flight software, and messages sent by the payload towards the ground).

Information gathered by these probes are used as an input to the detection algorithms. A representation of the architecture of NOS^3 and the location of the probes is given in Fig.2

It is important to note that, to have an IDS as accurate as possible, a standard model of commands sent from the ground which would represent real mission operation is needed. This can have a great impact on detection performance, as an unrepresentative model could hide weaknesses of the IDS or emphasize its strengths, therefore it is currently being investigated.

Using the first probe's information, an anomaly detection on packets characteristics is considered. The aim is to detect deviations from standard operations behavior, which could indicate an attack. A classic example is a Denial of Service (DoS) attack targeting the satellite, which consists in flooding

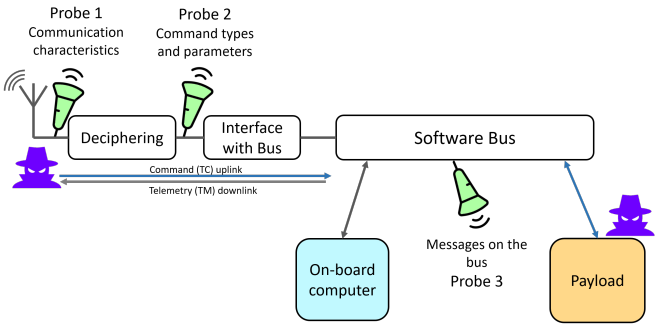

Fig. 2. Embedded probes in NOS^3 architecture

the satellite with packets coming from the ground. An abnormal activity could also be tied to the satellite's geographical location, if we have access to the ephemerises(for example, when it is not in visibility of a ground station). The challenging part is the definition of this standard behavior, which can be based on multiple metrics, and is tied to the model of standard operations aforementioned.

Another detection component using the second and third probe is considered and consists in a "TC firewall". This component is in charge of checking the validity of the TC identifier, and deploying a black list of dangerous TC. The aim is to address elementary attacks that use a single malicious TC, both sent from the ground or sent on the bus (if the payload was compromised).

The last detection component considered is in charge of analyzing the information gathered by the second and third probes to detect dangerous combinations of TC. All the applications embedded in the satellite interact by exchanging messages on the software bus present in NOS^3. Our detection strategy consists in both listening to all messages coming from the ground and to all messages exchanged in this bus. The aim is to identify anomalies that could reveal malicious actions. These actions could either be performed by a legitimate software component, on receipt of a malicious TC, or by a malicious payload. The challenge here is to have a model of legitimate messages exchanged on the bus, so that we can detect deviations in these communications. We are currently investigating different models, taking into account that our detection algorithms must be embedded in a platform that has limited resources in memory and computation. State-machine algorithms have been studied on message combinations, but due to the non-deterministic nature of the simulator behavior and bus communications, no satisfactory results were achieved so far.

V. Command Automation & Attacks

Another contribution of this work is the design and the implementation of a generic module aimed at carrying out various command scenarios on the NOS^3 platform. These scenarios can be legitimate, to simulate mission operations, as well as malicious, to send attacks. This module is essential so that we can assess the relevance of our detection algorithms. Attacks are also designed for the validation of the IDS, according to the threat model described previously.

A. Scenario Automation and Fuzzing

A framework allowing for the automated sending of TC is being developed, allowing for both benign and malicious scenarios to be conducted. Scenarios are pre-filled with selected TCs to be sent, and are described using JSON format. A fuzzing functionality is currently being developed, to automate the test of the NOS^3 simulator. The commands to be sent can be fully customized, field by field and some parameters of the commands can be randomized by means of the fuzzing module. Simple permutations will be investigated to improve fuzzing performance.

B. Attack library

An attack library based on NOS^3 is being developed within the CSS project. These attacks target both the ground segment and the space segment of the simulator, and also consider satellites alone and satellite constellations. Our attack contribution is centered on the space segment, and concerns DoS-type attacks that can deny service temporarily or permanently on the satellite. These attacks are supposed to target one single satellite.

VI. Conclusion

Satellite systems are becoming increasingly targeted by cyberattacks. Being strategic systems, and involved in many aspects of our daily lives, their security has become a major issue. The research work described in this paper aims at designing and implementing an intrusion detection system embedded in a satellite. This article has described the threat model and the attack scenarios considered as well as the architecture of the IDS that we currently investigate. This IDS will be implemented on the NOS^3 platform as well as the scenario component dedicated to carry out diverse attack scenarios, in order to assess the relevance of the IDS.

Acknowledgment

This research is supported by IRT Saint Exupéry (CSS Project).

References

[1] "Case study: Viasat attack," WebPage, Jun. 2022. [Online]. Available: https://cyberconflicts.cyberpeaceinstitute.org/law-and-policy/cases/viasat

[2] "Thales seizes control of esa demonstration satellite in first cybersecurity exercise of its kind," WebPage, Apr. 2023. [Online]. Available: https://www.thalesgroup.com/en/worldwide/security/press_release/thales-seizes-control-esa-demonstration-satellite-first

[3] M. Werremeyer, J. Williams, S. Wood, M. Walker, J. Ameen, and B. Kerley, "Hack-A-Sat: Four Years from the Cromulence Tech Team," in *2024 IEEE Aerospace Conference*, Mar. 2024, pp. 1–17, iSSN: 1095-323X. [Online]. Available: https://ieeexplore.ieee.org/document/10521107

[4] A. Schalk and D. Brown, "Detection and Mitigation of Vulnerabilities in Space Network Software Bus Architectures," in *2023 IEEE Aerospace Conference*, Mar. 2023, pp. 1–10, iSSN: 1095-323X. [Online]. Available: https://ieeexplore.ieee.org/document/10115986

[5] O. Driouch, S. Bah, and Z. Guennoun, "Intrusion detection system for CubeSats: a survey," in *2023 International Wireless Communications and Mobile Computing (IWCMC)*, Jun. 2023, pp. 596–601, iSSN: 2376-6506. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10182729

[6] ——, "CANSat-IDS: An adaptive distributed Intrusion Detection System for satellites, based on combined classification of CAN traffic," *Computers & Security*, vol. 146, p. 104033, Nov. 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404824003389

[7] "Nasa operational simulation for small satellites," WebPage. [Online]. Available: https://www.nasa.gov/nasa-operational-simulation-for-small-satellites/

[8] S. Gios, C.-H. Bertrand Van Ouytsel, M. D. Caribé, and A. Legay, "A vision on a methodology for the application of an Intrusion Detection System for satellites," in *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering*. Sacramento CA USA: ACM, Oct. 2024, pp. 2205–2209. [Online]. Available: https://dl.acm.org/doi/10.1145/3691620.3695314

[9] U. Uhongora, R. Mulinde, Y. W. Law, and J. Slay, "Deep-learning-based Intrusion Detection for Software-defined Networking Space Systems," *European Conference on Cyber Warfare and Security*, vol. 22, no. 1, pp. 639–647, Jun. 2023, number: 1. [Online]. Available: https://papers.academic-conferences.org/index.php/eccws/article/view/1085

[10] A. T. Azar, E. Shehab, A. M. Mattar, I. A. Hameed, and S. A. Elsaid, "Deep Learning Based Hybrid Intrusion Detection Systems to Protect Satellite Networks," *Journal of Network and Systems Management*, vol. 31, no. 4, p. 82, Sep. 2023. [Online]. Available: https://doi.org/10.1007/s10922-023-09767-8

[11] M.-J. Kang and J.-W. Kang, "A Novel Intrusion Detection Method Using Deep Neural Network for In-Vehicle Network Security," in *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, May 2016, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/document/7504089

[12] S. Khandelwal, E. Wadhwa, and S. Shreejith, "Deep Learning-based Embedded Intrusion Detection System for Automotive CAN," in *2022 IEEE 33rd International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, Jul. 2022, pp. 88–92, iSSN: 2160-052X. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9912045

[13] M. Ge, X. Fu, N. Syed, Z. Baig, G. Teo, and A. Robles-Kelly, "Deep Learning-Based Intrusion Detection for IoT Networks," in *2019 IEEE 24th Pacific Rim International Symposium on Dependable Computing (PRDC)*, Dec. 2019, pp. 256–25 609, iSSN: 2473-3105. [Online]. Available: https://ieeexplore.ieee.org/document/8952154

[14] Y. Fu, Z. Yan, J. Cao, O. Koné, and X. Cao, "An Automata Based Intrusion Detection Method for Internet of Things," *Mobile Information Systems*, vol. 2017, no. 1, p. 1750637, 2017, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1155/2017/1750637. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1155/2017/1750637

[15] M. Marchetti and D. Stabili, "Anomaly detection of can bus messages through analysis of id sequences," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 1577–1583.

[16] S. J. J. Généreux, A. K. H. Lai, C. O. Fowles, V. R. Roberge, G. P. M. Vigeant, and J. R. Paquet, "MAIDENS: MIL-STD-1553 Anomaly-Based Intrusion Detection System Using Time-Based Histogram Comparison," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 1, pp. 276–284, Feb. 2020. [Online]. Available: https://ieeexplore.ieee.org/document/8705276

[17] J. Thebarge, W. Henry, and G. Falco, "Developing scenarios supporting space-based ids," *ASCEND 2022*, Oct. 2022. [Online]. Available: https://arc.aiaa.org/doi/10.2514/6.2022-4219