An Intrusion Detection Process for an Avionic Embedded System

Aliénor Damien*†, Marc Fumey*, Eric Alata†, Mohamed Kaâniche†, Vincent Nicomette†
*Thales Avionics, Toulouse, FRANCE, Email: {firstname}.{lastname}@fr.thalesgroup.com
†LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, FRANCE, Email: {firstname}.{lastname}@laas.fr

***Abstract*—Security has become a critical concern for avionic actors due to the increasing threats targeting embedded systems and also to the recent evolutions of avionic systems. The threat surface of an aircraft has always been very light because of strong safety requirements. Therefore the aircraft transportation domain remains up to now, one of the most safe.**

Additional technical and organizational security measures begin to be deployed on recent airplanes to enhance protection against potential malicious threats. Nevertheless, continuous evolution of cyber threats imposes to reconsider the security of systems for the future. Several studies propose to introduce Intrusion Detection Systems (IDS) in embedded systems. This paper briefly describes the challenges raised by the introduction of such techniques in avionics systems and proposes a Host-based IDS process that is designed to fit avionics systems constraints.

I. Introduction

The threat surface of an aircraft has always been very tiny because of strong safety requirements [1], limited connectivity, and others specificities. As a consequence, attacks are difficult to perform and hard to reuse, even by experts having combined skills in cybersecurity and avionics.

However, the trend in aircraft systems is to make them connected and less expensive. As a matter of fact, Components Off-The-Shelf (COTS) will be used at a large scale, introducing wireless connectivity technologies on board for the passengers, increasing the sharing of resources between aircraft functions, etc. However, these evolutions will increase the threat surface that an attacker could potentially use to compromise the system.

Considering some recent attacks on embedded systems [2], [3], the navigability regulation has evolved [4]. Avionics actors have now to consider on board and ground infrastructure security, and some new aircrafts already have perimetric defense, by seperating the network in different domains [5], [6]. While aircrafts implement strong safety mechanisms that are historically designed to provide protection against accidental threats, to the best of our knowledge, they do not implement intrusion detection mechanisms in operation.

In the context of a PhD, two main research objectives are pursued : 1) investigate the efficiency of existing safety and dependability mechanisms when malicous threats occur during operation, and 2) develop complementary intrusion detection mechanisms to be deployed on embedded avionics platforms. These investigations will be supported by experimental analyses on a critical embedded systems.

This paper presents the research work of the first year of a PhD devoted to the introduction of intrusion detection systems (IDS) in avionic calculators. Section II focuses on existing IDS used in traditional information systems, and analysis their advantages and limitations with respect to the specific constraints inherent to avionic systems. Then Section III presents our approach to integrate a host-based IDS in an avionic context, and Section IV presents the overall process of this integration. Section V concludes and discusses future work.

II. IDS State of the Art & Avionic Domain

IDS are usually classified as Signature-based and Anomaly-based IDS [7]. The first ones look for attack patterns, and the other ones for deviations from a normal behavior. To apply these techniques on avionic systems, the following constraints have to be considered on the IDS:

- Real-time : must not disturb the real-time execution of the aircraft functions
- High safety level : must have a high criticality level and not introduce dependencies between applications
- Embedded : must not consume too many resources
- Maintenance : cannot be updated at each landing because of the high cost of a grounded aircraft for the airline
- Life time : must be efficient during at least 20 years
- Certification : its performances must correspond to its Development Assurance Level (DAL) [8].

Signature-based IDS need a database of known attacks (that does not exist today for avionic) with frequent updates, and are not able to detect new or sophisticated attacks. As a consequence, it would be unacceptable to implement this kind of IDS in avionic context. On the other hand, anomaly-based IDS may generate a lot of false alarms, and the cost of grounding a fleet due to a false alert would not be acceptable. Moreover, some difficulties may be raised with respect to certification if the algorithms implemented in the IDS are not deterministic. However, anomaly-based IDS present some interesting characteristics in avionic context. They are efficient to detect new attacks without requiring the update of an attack signature database [9]. Moreover, the modeling of the normal behavior of an avionic application can be performed with a better reliability than in IT context because the avionic environment is under strict control and is deterministic.

A few studies have been published about IDS in embedded systems. For instance, [10] highlight some related constraints and challenges. The use and implementation of IDS on

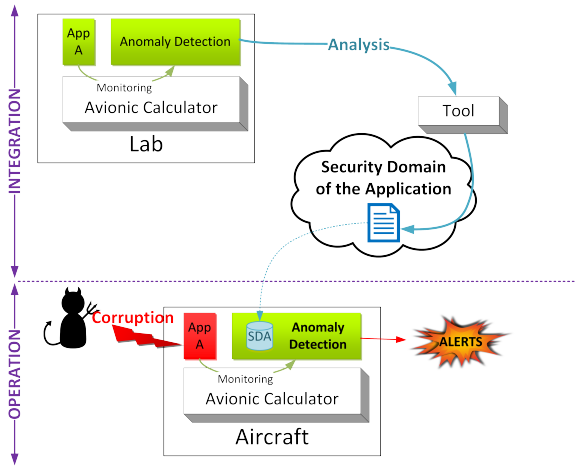


Figure 1. Detection Approach

Multi-core architectures for real-time embedded systems is investigated in [11]. Some studies propose hybrid IDS to take advantage of both signature-based and anomaly-based techniques [12]. Silvia & al [13] propose a network-based IDS in avionic system using network packets as input data. Our research focuses on integrating a Host-based IDS (HIDS) in the actual application development process, using avionic computer as a source of data.

III. Overall Approach

The goal of our research is to develop an anomaly based IDS that can be used to detect potential corruption of avionic applications during operation. The objective is to model and monitor avionics application's behavior during operation and raise alert when a deviation from expected behavior is detected. Even if an embedded application corruption is nowadays very difficult to perform, we consider the pessimist case where it may be corrupted, for instance in operation phase or during maintenance. Our main objective is to detect this corruption, thanks to an embedded HIDS. Possible solutions to respond to the raised alerts and to mitigate the impact of the detected intrusions are out of the scope of this paper.

Figure 1 summarises the main steps of the proposed approach which consists in monitoring in real time the usage of OS resources (like CPU usage, API calls, memory usage, ...) of the application and comparing this usage to thresholds defined in a "Security Domain of the Application" (SDA). The SDA is a set of rules characterizing the normal behavior of the application (for example, the application A should not call more than 10 API calls in one execution cycle). It is based on parameters chosen to detect security intrusions specifically, according to some specific feared threats. The SDA itself is defined during the development process, and loaded on the platform independently of the application. Deviations of application resources usage with respect to the specified SDA are detected by monitoring mechanisms, and can be reported to a diagnostic application in charge of establishing if the problem detected really corresponds to an attack.

Table I
Roles of Actors Involved in Conventional Development Process

Actor	Phase	Role
Application supplier	1-Application Development	Develop an application to provide an avionic functionality
Module integrator	1-Application Development	Allocate resources to the different applications
	2-Integration	Install applications on a module and perform the certification
Airline	3-Operation	Operate the aircraft

The following section illustrates how the proposed approach can be integrated in a traditional avionic developement process.

IV. HIDS development and Deployment Process

Table I describes the main phases of a conventional avionic process (application developement, integration, and operation) together with the main actors involved and their roles. The proposed HIDS process starts with the application delivery, at the end of the application development phase.

A. Application Delivery

Applications are delivered to the module integrator with documentation requested for the integration process. These documents should include information indicating the use of OS resources and services (resources usage contract, specification, source code, ...), used as inputs to establish the first version of the SDA.

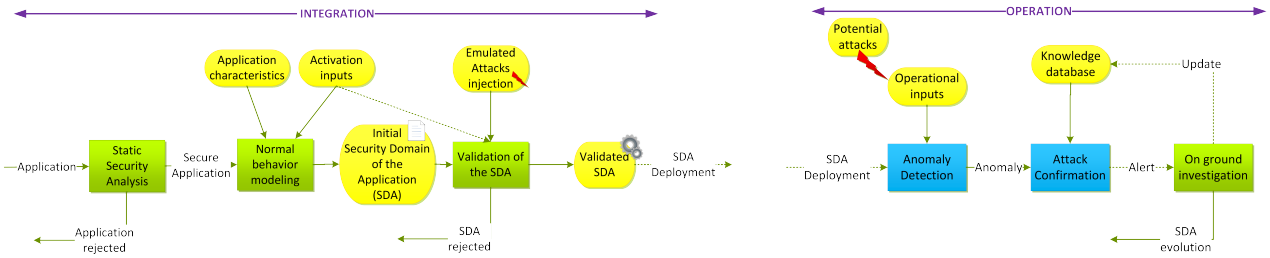
B. Integration

After the application delivery, the SDA is constructed during the integration phase as illustrated in Figure 2a.

1) Static Security Analysis: This analysis is performed on the binary code of the application. Usual security tests are executed, like vulnerability, antimalware or binary analysis. The compliance of the application with the documents provided by the application supplier is also checked. The formal acceptation of the application currently done in the integration process includes these static security checks.

2) Normal Behavior Modeling: The goal of this phase is to construct a preliminary SDA, representing the normal behavior of the application. It can be done directly from the application's documentation provided in the delivery phase (manually), or by running the application in a laboratory setup, by simulating its activation inputs, also provided by the application supplier. In this case, the elaboration of the SDA is done automatically by using machine learning technics.

3) Validation of the SDA: In this phase, the SDA is used to parametrize some sensors to monitor the characteristics of the application. Attacks are then injected in the application in a lab environment to test the efficiency of the detection. These attacks are injected using a tool emulating attack effects, by directly injecting data in memory, either in the binary code of the application or in some data manipulated by it.



(a) Integration (b) Operation

Figure 2. HIDS modules in overall process : (a) Integration and (b) Operation

If the previous SDA was constructed from the application's documentation only, the activation inputs can also be simulated to assess the accuracy of the SDA. If some attacks are not detected or if there are too many false alarms, the results are investigated to propose a new SDA. Finally, the SDA obtained after some iterations is considered to fit the normal correct behavior of the application and each deviation from this behavior is considered as possibly malicious.

C. Operation

Once the SDA is validated, it is deployed in the aircraft for the operational phase, as well as the application itself. Figure 2b shows the HIDS modules introduced in this phase.

1) Anomaly Detection: Using data from a set of sensors, the application behavior is monitored in operation to detect any behavioral anomaly, i.e., any deviation use of OS resources and service compared to the thresholds defined in the SDA. The existing sensors for safety requirements or maintenance process can be reused to get information, possibly without modifying the operating system. Some sensors may also be added to existing sensors to cover more threats.

2) Attack Confirmation & On Ground Investigation: Even if anomaly detection has many advantages to be used in avionic domain, the rate of false alerts and the potential need to update the model can be problematic. The attack confirmation module brings three functionalities to counter these issues:

- Anomalies characterization : Use a knowledge database to confirm a real attack or to exclude known false positives or safety-related anomalies
- Alerts sending : Send alerts with a degree of confidence to the crew and/or the ground depending on the anomalies characterization result
- Knowledge database updating : Update the knowledge database after investigation of non-confirmed anomalies on the ground

Comparing to a signature-based IDS which need very frequent updates, we consider that the knowledge database will need few updates, as it is only necessary for anomalies that are not already characterized.

V. Future Work

This paper discussed the challenges related to the use of traditional IDS techniques in the context of real-time critical avionics systems. We also presented the principles of an anomaly-based intrusion detection approach aimed at modeling and monitoring the behavior of an avionic application through a HIDS process adapted to avionic constraints. The integration of this approach into a conventional avionics application integration and operation process is also described.

In the future, we plan to define more precisely each step of the process presented, and to experiment it on a real use case. In particular, we are working on the definition of relevant parameters to monitor our SDA, the formalisation of the different classes of attacks to be emulated and injected, and the setup of an exprimental framework integrating the different modules of the proposed approach.

References

[1] SAE, ARP4754: *Certification Considerations for Highly-Integrated Or Complex Aircraft Systems*, Nov. 1996.

[2] *Jeep Hacking Incident Leads to Fiat Chrysler Recall of 1.4M Vehicles*, July 27, 2015. [Online]. Available from Claims Journal: http://www.claimsjournal.com/news/national/2015/07/27/264766.htm

[3] Calvin Biesecker, *Boeing 757 Testing Shows Airplanes Vulnerable to Hacking, DHS Says*, Aviation Today, 8 Nov. 2017

[4] EUROCAE WG-72 and RTCA SC-216, DO-326A/ED-202A: *Airworthiness security process specification*, Oct. 2010.

[5] ARINC Industry Activities, ARINC 664 P5: *Aircraft Data Network Part 5 Network Domain Characteristics and Interconnection*, Apr. 2005

[6] K. Netkachova, K. Müller, M. Paulitsch, and R. Bloomfield, *Investigation into a layered approach to architecting security-informed safety cases*, in IEEE/AIAA 34th Digital Avionics Systems Conference (DASC), 2015

[7] A. Pharate, H. Bhat, V. Shilimkar, N. Mhetre, *Classification of Intrusion Detection System*, International Journal of Computer Applications (0975 – 8887), Volume 118 – No. 7, May 2015

[8] RTCA and EUROCAE, DO-178B/ED-12B: *Software Considerations in Airborne Systems and Equipment Certification*, 1992

[9] S. X. Wu and W. Banzhaf, *The use of computational intelligence in intrusion detection systems: A review*, Appl. Soft Comput., vol. 10, no 1, p. 135, janv. 2010.

[10] F. M. Tabrizi and K. Pattabiraman, *Flexible Intrusion Detection Systems for Memory-Constrained Embedded Systems*, in Dependable Computing Conference (EDCC), 2015 Eleventh European, 2015.

[11] M.-K. Yoon, S. Mohan, J. Choi, J.-E. Kim, and L. Sha, *SecureCore: A multicore-based intrusion detection architecture for real-time embedded systems*, in Real-Time and Embedded Technology and Applications Symposium (RTAS), 2013 IEEE 19th, 2013, p. 21–32.

[12] G. Kim, S. Lee, and S. Kim, *A novel hybrid intrusion detection method integrating anomaly detection with misuse detection*, Expert Syst. Appl., vol. 41, no 4, p. 16901700, March 2014.

[13] S. Gil Casals, P. Owezarski and G. Descargues, *Generic and autonomous system for airborne networks cyber-threat detection*, in 32nd Digital Avionics Systems Conference (DASC), Syracuse, NY, Oct. 2013.