Increasing the Resilience of ATC systems against False Data Injection Attacks using DSL-based Testing

Aymeric Cretin
Dept. d'Informatique des Systèmes Complexes
FEMTO-ST Institute, UBFC, CNRS
Besançon, FRANCE
aymeric.cretin@femto-st.fr

Bruno Legeard*†, Fabien Peureux*†, Alexandre Vernotte*
Dept. d'Informatique des Systèmes Complexes
*FEMTO-ST Institute, *UBFC, *CNRS, †Smartesting
Besançon, FRANCE
{blegeard, fpeureux, avernott}@femto-st.fr

***Abstract*—Within aircraft communication, due to the unauthentication and unencryption of the Automatic Dependent Surveillance-Broadcast (ADS-B) protocol, eavesdropping and broadcasting fake ADS-B messages is straightforward. As a consequence, attackers can perform False Data Injection Attacks (FDIA) on the ADS-B system, such as ghost aircraft injection or flooding, leading to unexpected but potentially devastating consequences. To increase the resilience of Air Traffic Control (ATC) systems against such attacks, this paper presents a framework under development that aims to generate FDIA-based test scenarios, which can be used as test cases to evaluate and improve the robustness of ATC systems. This test generation framework uses a Domain Specific Language (DSL) to specify FDIA-based test strategies in order to falsify legitimate ADS-B recordings. Such generated altered ADS-B recordings are finally executed on ATC systems to evaluate its resilience against FDIA. The paper details this process and introduces early results and future work.**

Keywords*—Air Traffic Control, ADS-B protocol, False Data Injection Attacks, Automated Test Generation.

I. Introduction

The world of air transport is facing new challenges as the traffic load keeps growing steadily[1]. With an increasingly congested airspace, Air Traffic Control (ATC) must improve accordingly in terms of simultaneously handled aircraft as well as positioning accuracy. The Automatic Dependent Surveillance-Broadcast (ADS-B) protocol is currently being rolled out in an effort to reduce costs and improve aircraft position accuracy [1]. Communication via ADS-B consists of participants broadcasting their current position and other information periodically (a.k.a. a beacon) in an unencrypted message [2]. The ADS-B protocol was not designed with security in mind, since securing ADS-B communication was not a high priority during its specification. As a consequence, anyone with the right equipment can listen and emit freely. For instance, there is a market for equipping private aircraft with ADS-B transponders using a smartphone and a dongle[2]. While it is not the only means for Aircraft tracking — other protocols like VHF, CPDLC, ACARS are also used in conjunction of Primary Surveillance Radar (PSR) —, ADS-B plays a central role in the current shift regarding aircraft position collecting from radar systems to a global navigation satellite system [3]. Hence there is a strong need to improve its overall security. However, because of the inherent properties of the protocol, current solutions for securing ADS-B communications are only partial or involve an unbearable cost [4]. Therefore ATC should be made more secure by strengthening its logic, but the ability to differentiate attacks from real critical situations still remains a challenge that must be tackled.

To reach this goal, the current paper proposes a DSL-based testing approach and tool capable of creating, modifying and deleting recorded legitimate ADS-B messages in an intelligent manner. Concretely, it provides a test framework for ATC experts to test their system against potential security and safety anomalies related to False Data Injection Attacks on ADS-B messages. Its application is two-fold: first to assess the current cyber security of ATC systems, and second to periodically measure expected cyber security improvements.

II. State of the Art

This paper focuses on complex and elaborate attacks against air traffic communications related to the ADS-B protocol. Because of the very nature of ADS-B, anyone with the necessary (but relatively cheap) equipment can perform illegal operations such as listening, blocking, creating and modifying ADS-B messages. Eavesdropping (i.e. listening) and Jamming (i.e. blocking) are the most straightforward attacks. They are also considered as low-level, simple technical attacks and thus not specific to ADS-B: they are therefore outside the scope of this work. Conversely, more elaborate attacks require a deep understanding of the system, its protocol(s) and its logic, to covertly alter (by injecting falsified beacons and deleting genuine ones) the consensus reached by ground stations and aircraft regarding the Recognized Air Picture (RAP). Such attacks are much more complex to achieve than e.g., jamming because the logic of the communication flow must be preserved. This type of attack is commonly referred to as *False Data Injection Attack* (FDIA) in the literature.

FDIAs were initially introduced in the domain of wireless sensor networks [5]. A typical scenario consists of an attacker who first penetrates a sensor network, usually by compromising one or several nodes, and thereafter injects false data

[1]http://www.boeing.com/commercial/market/current-market-outlook-2017/
[2]https://www.uavionix.com/products/skybeacon/

reports to be delivered to the base station. This can lead to the production of false alarms, the waste of valuable network resources, or even physical damage. Active research regarding FDIAs has been conducted in the power sector, mainly against smart grid state estimators [6], [7]. It shows that such attacks can lead to power blackouts but also directly affect electricity markets [8]. The current paper proposes to extend the definition of FDIA to the domain of ATC by making the analogy between nodes and base stations in a wireless sensor network, and aircraft and ground stations respectively in Air Transport.

The means of the attacker to conduct such attacks against ADS-B communication have already been detailed in previous work [9]. Considering the attacker has the necessary equipment, he can perform three malicious operations: (i) *message injection* which consists of emitting non-legitimate but well-formed ADS-B messages, (ii) *message deletion* which consists of physically deleting targeted legitimate messages using destructive or constructive interference, and (iii) *message modification* which consists of modifying targeted legitimate messages using overshadowing or bit-flipping. It is worth mentioning that message deletion should not be mistaken for jamming, as jamming blocks all communications whereas message deletion drops selected messages only.

These three techniques allow for the execution of several attack scenarios [10], as detailed below:
Ghost Aircraft Injection. The goal is to create a non-existing aircraft by broadcasting fake ADS-B messages on the communication channel.
Ghost Aircraft Flooding. This attack is similar to the first one but consists of injecting multiple aircraft simultaneously with the goal of saturating the RAP and thus a denial of service of the controller's surveillance system.
Virtual Trajectory Modification. Using either message modification or a combination of message injection & deletion, the goal of this attack is to modify the trajectory of an aircraft.
False Alarm Attack. Based on the same techniques as the previous attack, the goal is to modify the messages of an aircraft in order to indicate a fake alarm. A typical example would be modifying the squawk code to 7500, indicating the aircraft has been hijacked.
Aircraft Disappearance. Deleting all messages emitted by an aircraft can lead to the failure of collision avoidance systems and ground sensors confusion. It could also force the aircraft under attack to land for safety check.
Aircraft Spoofing. This scenario consists of spoofing the ICAO of an aircraft through message deletion and injection. This could allow an enemy aircraft to pass for a friendly one and reduce causes for alarm when picked up by PSR.

Designing, implementing and executing test scenarios on the ATC systems to assess the behavior of ATC systems in case of such alterations of ADS messages by support of a dedicated tooling is the primary goal of our work. To the best of our knowledge, there are no direct previously published works addressing this topic. Barreto et. al. propose a framework that allows for the simulation of an entire air traffic environment (aircraft, radio relay, network infrastructure, etc.) [11]. They perform FDIAs in the simulated environment to evaluate the attack impact on each component. The approach provides a lot of information on how components react to an FDIA. Nevertheless, implementing all network behavior of a scenario requires a lot of effort and the approach does not allow for the concretization of the simulated attacks on production ATC software. Conversely, the presented approach focuses on the reaction of software that interprets ADS-B and displays information to the ATC system.

III. Objective of Research

The contribution presented in this paper is part of an ongoing research initiative about applications of automated test generation techniques that are driven by formalized test patterns and based on behavioral models — i.e. Model-Based Testing — for security testing. A first axis of research focuses on vulnerability testing of Web and mobile applications to detect technical vulnerabilities, such as SQL injections, but also logical vulnerabilities, such as Insecure Direct Object References [12]. A second axis focuses on the detection of FDIAs in air traffic control systems, and more precisely on the ADS-B protocol [13]. Experimentation results demonstrated the feasibility of producing attack scenarios targeting ADS-B communications. There were also weaknesses to the initial approach [14], inherent to Model-Based Testing techniques in general, especially the required effort related to modeling activities (i.e. creation and maintenance of the MBT models) and the gap between MBT technologies with current tester skills. The aim of the approach introduced in this paper is to overcome the weaknesses that the first approach suffered from.

To meet these expectations, we propose to use legitimate ADS-B recordings as input to avoid the creation and the maintenance of MBT models. We also propose to create a Domain-Specific Language (DSL) that would allow security experts to easily specify alteration directives, i.e. how legitimate ADS-B messages should be modified / deleted and what illegitimate messages should be created in order to perform elaborate attacks, as discussed in Section II. Concretely, the objective of research (RO) is defined as follows:

RO: How to automate resilience testing of ADS-B based ATC systems against FDIAs?

This objective constitutes the heart of the studied problem, and can be split into three more specific research questions:

RQ1: To what extent can test design using a DSL-based testing be efficiently supported for FDIA?
While it is indubitable that FDIAs can bear devastating consequences on air traffic, these attacks are also known to be quite complicated to execute [6]. Therefore, being able to design test cases capable of detecting vulnerabilities to such attacks in ATC systems constitutes a great challenge.

RQ2: To what extent can test execution and verdict assignment be automated?
Offering an efficient way to design test cases is important but it is hardly valuable if execution and verdict assignment are not automated. Indeed, software testing in general is known to be a time consuming activity, and achieving test execution and verdict assignment automation is an always pursued goal.

RQ3: To what extent can the DSL be extended to other types of ATC data?
The DSL in its initial form only targets the ADS-B protocol. However, it ought to be extendable to other protocols that

transport similar information. Extending the DSL's expressiveness to be compatible with other data exchange protocols, such as Cat062 (System Track Data) from the EUROCONTROL ASTERIX standards [15] or the Flight Information eXchange Model (FIXM), is of strong interest: it would make it possible to simulate multi-protocols FDIAs.

IV. Proposed Approach

To increase ATC systems' resilience against FDIAs, the proposed approach combines DSL-based testing with an alteration generation and selection method. This section first introduces the principles of this approach, then presents early achievements in terms of DSL expressiveness and related alteration generation algorithms, and finally details an early implementation of the test framework.

A. Overall Process

The overall process of the proposed approach is depicted in Figure 1. It consists of six activities:

① **Data acquisition**. This is about collecting legitimate ADS-B recordings, obtained either from the internet (e.g., via OpenSky Network) or by eavesdropping on the RAP using an antenna (in practice, we use the Kinetic SBS-3 receiver).

② **Scenario Design**. ATC experts use the DSL to design attack scenarios. A scenario specifies the alterations necessary to perform an FDIA. It can be designed graphically to target a specific air traffic recording, or textually using the DSL to target a set of recordings (thus allowing for combinatorial test design). This paper centers on the latter method.

③ **Alteration Generation**. Because DSL-based scenarios can be applied to multiple air traffic recordings, a dedicated algorithm interprets scenarios and produce a set of tailored alteration directives for each ADS-B recordings given as input.

④ **Concretization**. Another algorithm takes as input an original recording and its associated alteration directives to produce an altered recording in the ATC system input format (e.g., Socket Data Format).

⑤ **Injection**. The obtained altered air traffic recordings are (sequentially) sent to the ATC system under test.

⑥ **Assessment**: ATC system output is compared with the expected results defined during scenario design, and a test verdict (pass or fail) is assigned.

B. Early Achievements

A DSL is a language that provides a particular expressiveness related to a specific domain [16]. For the current approach, the proposed DSL is specific to the design of FDIA-based test scenarios that target ATC protocols. By relying on the ATC-specific terminology, it gives ATC experts the ability to define FDIA-related attack scenarios that can be applied to a set of air traffic recordings. An attack scenario consists of combining basic alterations. A basic alteration is composed of four parts:
Action: type of action to perform, i.e. creation, modification or deletion.
Scope: conditions in which the targeted aircraft should be affected. Conditions can be spatial, temporal, or related to the properties of aircraft such as ICAO, callsign, altitude, etc.

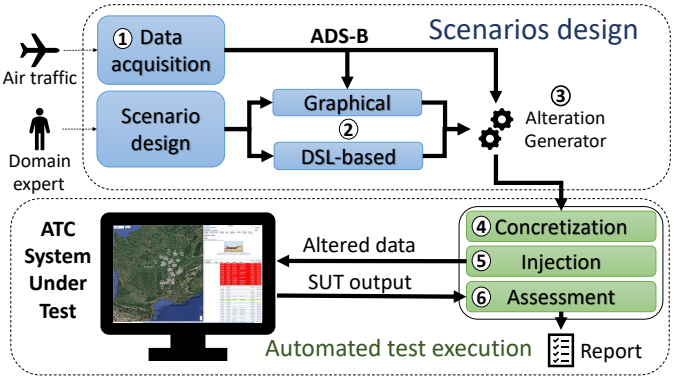

Fig. 1: Process of the proposed approach

Parameters: parameter values to be changed or set such as altitude, longitude and latitude, squawk code, etc.
Assertions: how the ATC system should react to the alteration.

```
1 create plane from 20 seconds until 320 seconds
2 with_values ICAO = "39AC47"
3 and GROUNDSPEED = 102.2 assert "GS too low"
4 and CALLSIGN = "SAMU25"
5 global assert "Fake aircraft detected"
```

Fig. 2: Example of a Ghost Aircraft Injection

Figure 2 shows an example of a *Ghost Aircraft Injection* scenario. The instruction **create** (line 1) constitutes the *Action* part which, in this example, is a message creation. The *Scope* part (line 1) is temporal only: a plane should be created from 20 **seconds** to 320 **seconds** according to the start of the input recording. The *parameters* part contains three parameters to set (lines 2-4), namely **ICAO**, **GROUNDSPEED** and **CALLSIGN**. Finally, the *Assertion* part is composed of a parameter-specific assertion (line 3) and a global assertion (line 5) related to the entire scenario. It means that, for the test case to pass, the ATC system should return a message that the ground speed of that particular aircraft is too low, and thereafter another message claiming it detected a fake aircraft. One single aircraft is created with this example. However, the DSL syntax includes loop structures, list structures and variables (to change parameters values between loop iterations) making it possible to, for instance, iterate the above example to perform *Ghost Aircraft Flooding*. This task is achieved by the alteration generator. Therefore designing of FDIA-based test scenarios using the DSL has the advantage of bringing automation, as such test generation patterns can be applied to many air traffic recordings.

As suggested in *RQ3*, the DSL aspires to be compatible with multiple surveillance protocols, taken that they transport information similar in nature. As of now however, it is tailored to the ADS-B protocol. This can be done by extending the language, for instance by adding new types of *parameters* and *criteria*. Another potential solution to investigate would be to introduce a Domain Specific Modeling Language (DSML) in the third activity to serve as pivot between the DSL and the targeted protocols. It would capture all possible alteration directives after interpretation of the DSL, and each protocol would have its own concretization module. As a consequence,

the DSL's grammar would not have to be extended since the specialization to ATC protocols would happen at a later stage, after instantiation of the DSML.

C. Early Prototype

A first prototype of the approach has been implemented. Its user interface is depicted in Fig. 3. Whilst not (yet) complete in terms of alteration types and protocols, the prototype does provide an end-to-end tool chain, and therefore demonstrate the practicality of the approach. The prototype currently allows users to design FDIA-based test scenarios for the ADS-B protocol, both graphically and textually using the DSL. Then, they can simulate these scenarios on an ATC system. However, test cases generated with the prototype are not yet performed on a real ATC system, but altered recording are injected in Virtual Radar [17], a software program that listens for different ATC protocols and displays aircraft information and positions on a map when it receives ADS-B socket data stream.

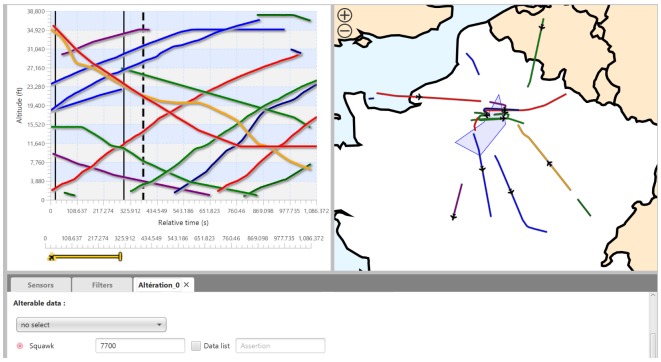

Fig. 3: Graphical User Interface of the prototype

For graphical scenario design, a BST file can be imported in order to present all ADS-B traces graphically. As shown in Fig. 3, each ADS-B trace is represented by a trajectory on a map (see right side of the figure), and a graph shows the altitude variations of aircraft according to time (see left side of the figure). For each ADS-B trace, the user can choose what kind of action s/he wants to perform. Most properties of ADS-B messages can be modified: altitude, longitude, latitude, squawk, ICAO, ground speed or call sign. Graphical-based test design has the advantage to be intuitive and easy-to-use. However, and contrary to DSL-based design, this method cannot be automated. Indeed, a graphical scenario is tied to an ADS-B recording and must be recreated to be applied to another record. Rather than rely on graphical creation of FDIA-based test scenarios, the prototype offers the user to textually design attacks from the DSL presented in Sect. IV-B. On the other hand, the DSL is quite close to natural language and includes ATC-related terms, allowing users (i.e. ATC security experts) to quickly accommodate to this textual-based design.

All in all, graphical and DSL-based designs lead to the same result: the generation of alteration directives. These instructions are applied to ADS-B messages and an altered ADS-B recording is obtained as output. The connection to inject altered ADS-B recordings in the ATC system is done through a simple RJ45 connector. The system only needs to be on the same network than the PC of the prototype user. Early promising experiments have demonstrated the feasibility of the approach and forecasted its effectiveness and efficiency.

V. Conclusion and Future Work

This paper describes an innovative DSL-based test framework and related process to capture and alter ADS-B communication recordings. The objective is to evaluate and improve the resilience of ATC systems against FDIA attacks. An early but end-to-end prototype implementing this approach has also been introduced. As a future work, the prototype will be enhanced, especially regarding the automation of the verdict assignment, but also by covering more ATC protocols including EUROCONTROL ASTERIX standards [15] and the Flight Information eXchange Model (FIXM).

References

[1] A. Smith, R. Cassell, T. Breen, R. Hulstrom, and C. Evers, "Methods to provide system-wide ADS-B back-up, validation and security," in *25th Digital Avionics Systems Conference*. IEEE, 2006, pp. 1–7.

[2] I. Martinovic and M. Strohmeier, "Security of ads- b: State of the art and beyond," *DCS*, 2013.

[3] P. Brooker, "Sesar and nextgen: investing in new paradigms," *The Journal of Navigation*, vol. 61, no. 2, pp. 195–208, 2008.

[4] M. Strohmeier, M. Schfer, R. Pinheiro, V. Lenders, and I. Martinovic, "On perception and reality in wireless air traffic communications security," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 6, pp. 1338–1357, 2017.

[5] M. Ma, "Resilience against false data injection attack in wireless sensor networks," in *Handbook of Research on Wireless Security*. IGI Global, 2008, pp. 628–635.

[6] G. Dan and H. Sandberg, "Stealth attacks and protection schemes for state estimators in power systems," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*. IEEE, 2010, pp. 214–219.

[7] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, no. 1, p. 13, 2011.

[8] L. Xie, Y. Mo, and B. Sinopoli, "False data injection attacks in electricity markets," in *Smart Grid Communications (SmartGridComm), First International Conference on*. IEEE, 2010, pp. 226–231.

[9] M. Strohmeier, "Security in next generation air traffic communication networks," Ph.D. dissertation, Oxford University, 2016.

[10] M. Schäfer, V. Lenders, and I. Martinovic, "Experimental analysis of attacks on next generation air traffic communication," in *International Conference on Applied Cryptography and Network Security*. Springer, 2013, pp. 253–271.

[11] A. B. Barreto, M. Hieb, and E. Yano, "Developing a complex simulation environment for evaluating cyber attacks," in *Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*, vol. 12248, 2012, pp. 1–9.

[12] A. Vernotte, "A pattern-driven and model-based vulnerability testing for web applications," Ph.D. dissertation, Univ. de Franche-Comté, 2015.

[13] J. Botella, P. Cao, C. Civeit, D. Gidoin, and F. Peureux, "Model-based test generation of aicraft traffic attack scenarios using ads-b standard signals," 1st User Conference on Advanced Automated Testing (UCAAT), 10 2013.

[14] F. Bouquet, F. Peureux, and F. Ambert, "Model-based testing for functional and security test generation," in *Foundations of Security Analysis and Design VII*. Springer, 2014, pp. 1–33.

[15] Eurocontrol, *All Purpose Structured, Eurocontrol Radar Information Exchange (ASTERIX)*, Eurocontrol, Apr. 2012.

[16] A. Van Deursen and P. Klint, "Domain-specific language design requires feature descriptions," *Journal of Computing and Information Technology*, vol. 10, no. 1, pp. 1–17, 2002.

[17] L. K. A. MSc, *Virtual Radar - Using the SBS-1er and Basestation Software*. Lulu.com, 2011.