Issues in Information Systems
Volume 24, Issue 1, pp. 170-185, 2023

DOI: https://doi.org/10.48009/1_iis_2023_115

Detecting ADS-B replay cyberattacks in the national airspace system

George Ray, *Shepherd University, gray@shepherd.edu*
Jeffrey Ray, *UMBC, jeffray@umbc.edu*

Abstract

This paper investigates the cybersecurity risks in the National Airspace System (NAS) with the introduction of Automated Dependent Surveillance Broadcast (ADS-B) equipment, which has been required for regulated airspace since 2020 and will be essential part of the Next Generation (NextGen) Air Transportation System (ATS) infrastructure. It details the national security implications of the cybersecurity vulnerabilities in the ADS-B avionic system and reviews proposed mitigations. One of the possible attacks on ADS-B is the replay attack and the authors propose and test a method to detect such an attack using cosine similarity. To validate the approach, the authors developed a computer-based ADS-B system using an RTL-Software Defined Radio (SDR) receiver to capture over 14 million live ADS-B transmissions from a region northwest of the Dulles International Airport. These readings were organized into a relational database for analysis. A data-driven detection algorithm was implemented with parallelism through shared memory and a live feed of ADS-B traffic interspersed with previously recorded message-sets as replay attacks. The system successfully detected various replay attack scenarios. Suggested mitigation measures for countering replay attacks are presented.

Keywords: data mining, cosine similarity, ADS-B, Mode-S, WAAS, GPS, multilateration.

Introduction

The Government Accountability Office (GAO) presented a report to the U.S. Congress on the cybersecurity risks to the National Airspace System (NAS) from vulnerabilities in avionics systems (GAO, 2020). The Federal Aviation Administration (FAA) is charged with the management of risk in the system to ensure the integrity and availability of the air space. Avionics equipment is essential for safe flight operations and is part of the networks that share data with flight crews, other aircraft and the Air Traffic Control (ATC) system. In its report, the GAO found that the FAA has not fully assessed the avionics risks in the NAS and that one of those avionic systems, Automatic Dependent Surveillance-Broadcast (ADS-B) posed significant risks.

The GAO (2018) advised the FAA and the Department of Defense (DoD) on the national security implications of ADS-B. ADS-B broadcasts operational data over networks in the NAS to other aircraft and to ATC. It broadcasts aircraft tracking information such as position, direction, altitude and velocity.

This paper will report on the nature of the information vulnerabilities posed by ADS-B and list mitigations proposed in the literature. The research will further detail the setup of a Software Defined Radio (SDR) system to receive ADS-B transmissions from aircraft and insert them into a data store for analysis of the factors that affect replay cyberattacks on ADS-B. A replay attack records ADS-B messages and replays them later with the intent of disrupting airspace surveillance. A detection system is developed that addresses the factors affecting ADS-B attacks and results are presented. Extending the implemented solution beyond research and into production is then addressed.

A security concern with ADS-B is that the protocol is unencrypted and unauthenticated (GAO, 2020; GAO, 2018; Ronen & Ben-Moshe, 2021; Wang, Zou, & Ding 2020). There are challenges with changing a protocol or algorithms in already certified avionics systems as such change can take years and cost millions of dollars to modify even a single line of code (GAO, 2020). In addition, misconfigurations can undermine the airworthiness of aircraft and due to the nature of software bugs, there may be latent defects not discovered until later, and only under certain conditions.

Automated Dependent Surveillance Broadcast

Current radar surveillance and multilateration technologies are having difficulties meeting the increasing needs of the Air Traffic Management (ATM) system (Wang, Zou, & Ding 2020). Multilateration uses the timed difference of arrival at several stations to determine the position of a mobile object. ADS-B is attractive as an alternative protocol for air surveillance because of its accuracy, broad coverage, and data sharing capability. It's an open architecture that broadcasts its message in a simple format and so becomes an opportunity for hackers, who have an increasingly sophisticated arsenal of attack methods. ADS-B is vulnerable to eavesdropping, jamming, modification, injection, and replay/playback attacks. An ADS-B system cannot identify authorized users from unauthorized and is therefore vulnerable to spoofing attacks (GAO, 2020; GAO, 2018; Ronen & Ben-Moshe, 2021; Wang, Zou, & Ding 2020).

Compared with current surveillance radars used in air traffic control, ADS-B has an easier and lower cost implementation and produces highly accurate data (Leonardi & Sirbu, 2021). Because ADS-B ground stations can be installed in any location, they do not have the problem of uncovered spots, present in the current system, for locations where ground radar cannot be installed. Furthermore, the better location accuracy of ADS-B permits smaller aircraft separations resulting in higher capacity in the NAS.

Mode-S transponders provide ATC aircraft identification, altitude information, and permit data communication via ADS-B messages. Mode S is a data link service that transfers data between nodes in a network. The Wide Area Augmentation System (WAAS)/Global Positioning System (GPS) is the only system that meets the ADS-B specification, and the NAS has over 154,942 WAAS equipped aircraft (Nelson, 2016). The WAAS is a satellite-based air navigation system developed by the FAA to augment GPS position accuracy, availability, and signal integrity, which enables improved enroute position information and precision approaches at airports without Instrument Landing System (ILS) equipment. As an onboard GPS receiver gathers GPS position information, a WAAS transceiver captures GPS corrections, applies them to the GPS data, and computes an accurate aircraft position that is broadcast to ATC via Mode-S ADS-B messages. WAAS/GPS can be part of a Flight Management System (FMS), another WAAS/GPS navigator, or a standalone WAAS/GPS sensor employed for use with ADS-B.

An aircraft equipped with ADS-B In receives information from other aircraft, ground control, and position information from onboard receivers. Aircraft equipped with a 1090 MHz (Mode S) transponder gather WAAS/GPS data from onboard systems, such as Inertial Reference Systems (IRS), Altitude Heading Reference Systems (AHRS), and Air Data Systems (ADS), process the information, and then transmit it (Elofson, Redondo, Francetic, et al, 2018). An aircraft equipped with ADS-B Out sends WAAS/GPS derived position and velocity data from the aircraft systems through an ADS-B Mode-S transponder out to ATC and other aircraft.

The system provides accurate location and flight path information resulting in safer operations, improved flight paths and reduced costs. In addition, the system improves air traffic control, which can collect and organize altitude, heading and speed of aircraft in its traffic space for conflict detection and resolution. ADS-B uses Global Navigation Satellite System (GNSS) position information along with other data to deliver its output about the aircraft and its spatial attributes, which it broadcasts via radio communications (Ronen & Ben-Moshe, 2021).

ADS-B transponders broadcast messages using the radar Mode S protocol. Mode S was a successor to the original ATC radar contact system; Mode S addressed overloading that occurred in ATC Radar Beacon Systems (ATCRBS). Mode S includes selective interrogation that can be directed to a single aircraft rather than to all aircraft. Mode S-ES is an extension of Mode S in which the packet length was increased from 56-bits to 112-bits allowing for inclusion of GNSS information in the messages. Mode S-ES, also known as 1090ES, is the standard for ADS-B (Burfeind, 2020). ADS-B transmits 120 microsecond messages that have a data block of 112 bits that use 112 microseconds with an 8-microsecond synchronization preamble. The Mode S broadcast message has two formats: a 56-bit short message and the 112-bit extended message format used by ADS-B (McCallie, Butts & Mills, 2011; Kožović & Durdevic, 2021). The fields in the message structure are as follows:

- 8 microsecond preamble for synchronizing the transmitter and receiver
- First 5 bits are the format encoding the message
- Second 3 bits is the transponder capability
- Third 24 bits is a unique aircraft identifier
- The next 56 bits refer to field data such as call sign, position, etc.
- The last 24 bits are a parity code

The 120 microsecond RF signal is converted to a digital stream using an analog-digital converter, and the resulting data structure is an ADS-B frame that is 112 bits long and consists of five main parts, shown in the Figure 1 message structure (McCallie, et al 2012; Sun, 2021). The data flow in the ADS-B process is for the aircraft equipment to compute its position using GNSS along with onboard sensors such as altimeters and then broadcast a state message to ground ADS-B units as well as other aircraft ADS-B units.

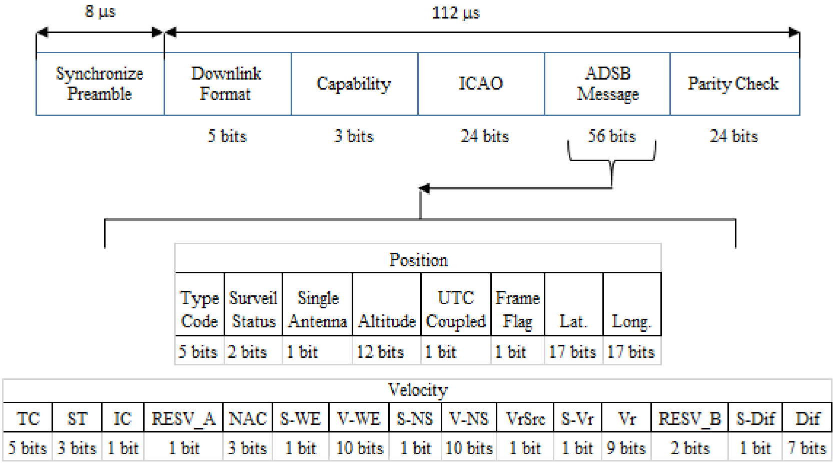

Figure 1: ADS-B Data Structure

Literature Review

Literature on ADS-B Vulnerabilities and Potential Cyberattacks

ADS-B has vulnerabilities including its dependency on GNSS and that it is based on a simple protocol that is unencrypted and unauthenticated. There are concerns about ADS-B vulnerability to jamming and spoofing. The types of potential cyber-attacks on the ADS-B system are shown in Table 1 (Alghamdi, Alshrahani & Hamza, 2018; Kožović & Durdevic, 2021). In addition to malicious hackers, military experiments on GPS jamming and spoofing are on the rise, and this affects ADS-B in commercial flights (Ronen & Ben-Moshe, 2021).

Table 1: ADS-B Attack Types

Attack Type	Description
Eavesdropping	Listening to broadcast messages - difficult to detect without encryption
Jamming	Denial of service attack through a jammer RF signal to prevent communication
Injection	Transmitting malicious messages, e.g. creating fake aircraft with menacing trajectories
Deletion	Using signal interference to garble certain portions of a message or an entire message
Modification	Bit flipping in transmitted messages to change the message meaning
Spoofing/Replay	Attacks to create false position or other spatial information

Jamming is aimed at the receiver system and consists of a high-powered RF signal broadcast at 1090 MHz, the frequency used by ADS-B to transmit messages from the ADS-B Out signal on the aircraft to the ADS-B In signal on the ground, to ATC systems, and other aircraft. The purpose of jamming is to disable the 1090 MHz signals sent by the ADS-B Out system.

Spoofing, on the other hand, attempts to introduce false messages into the 1090 MHz traffic so that ADS-B In systems process incorrect information. This can result in spurious positions for actual aircraft and the introduction of fictitious aircraft; both of which can result in failure of traffic management by flooding the airspace with ghosts or falsely representing aircraft flight paths as being on a collision course (Ronen, et al, 2021). Spoofing attacks can be executed with inexpensive Realtek RTL-SDR equipment and a PC. Rawat, Chakrawarti, Vyas, Gonzáles, Sikarwar, and Bhardwaj (2023) note that the first phase in a cyberattack on a digital architecture is a reconnaissance of the target to identify patterns of operation and vulnerabilities. Using an RTL-SDR device would enable such a reconnaissance on ADS-B, which lacks authentication or encryption.

Literature on Mitigations to the Cyberattacks

Mitigations have been proposed for the various ADS-B vulnerabilities. Table 2 provides a summary of this section. To protect against eavesdropping and injection, proposals have been made to modify the protocol to include encryption and authentication. Alghamdi, Alshrahani, and Hamza (2018) reviewed techniques to encrypt ADS-B traffic and found that Staged Identity-Based Encryption (SIBE) was the most effective method. The advantages of this method are superior key management for frequent encryption.

As noted by the GAO (2020), modifying a production avionics protocol is both costly and incurs the risk of latent defects. Furthermore, the FAA asserts that adding these security measures to ADS-B undermines the fundamental idea and benefit of ADS-B (Thurber, 2012). So, modifying the current ADS_B system to support authentication and encryption comes with significant cost.

Table 2: Proposed Mitigations

Attack Type	Mitigations	Potential Mitigation Risks
Eavesdropping	Encryption, authentication	Cost, latent defects
Jamming	Enhance rcvr hw, antenna nulls,array signal processing	Cost
Injection	Encryption, authentication, TDOA, MLAT	MLAT ground only
Deletion	Enhance rcvr hw, Synch'd rcvr networks	UTC sync availability
Modification	TDOA, Synch'd rcvr networks	UTC sync availability
Spoofing/Replay	Array signal processing, Encryption, authentication,	Cost, latent defects

Authentication and encryption will not help with jamming and message deletion. Anti-jamming approaches must be taken that include enhancing receiver hardware as well as utilizing multiple stations in a network that has measures to negate jamming and message deletion. In addition, the interfering signal can be reduced in strength by adjusting the antenna so that the offending signal is in an antenna null (Leonardi & Sirbu, 2021). A null in radio electronics is a minimum in a selected signal; a direction for the antenna where a signal fades abruptly (Lux & Schaefer, 2005). A signal from a malicious source can be reduced, possibly ignored, by directing the antenna so the malicious source is in an antenna null.

Leonardi and Sirbu (2021) propose a crowd sensor network to detect message tampering and injection by estimating Time Difference of Arrival. They suggest a detection approach that leverages synchronized receivers in global ADS-B networks such as FlightRadar24 and OpenSky networks of ADS-B receivers. They assert that their approach can detect ADS-B message injection, jamming and spoofing attacks without needing to resolve difficulties found in multilateration.

Ronen and Ben-Moshe (2021) propose an Internet of Things (IoT) technology solution to protect from jamming and spoofing attacks. ADS-B incorporates GNSS, a barometric altimeter and a transponder, which all have cyber security vulnerabilities. Goward (2019) reports on a passenger aircraft that nearly crashed into a mountain because of GNSS outages.

Ronen and Ben-Moshe (2021) introduce a mitigation using the LoRa 2.4GHz that can be used to protect not only against ADS-B spoofing but also GNSS spoofing. LoRa is a protocol using long range radio to connect Internet of Things devices. Currently there has been growth in Low-Power Wide Area Networks (LPWAN) that use long-range radio communications and a network configured to connect to the Internet. A disadvantage of LoRa is that it shares unlicensed bands with other devices using the same frequency.

Ronen, et al (2021) used a simulation to validate and verify the approach because conducting cyberattacks is a violation of law. They found that their approach has an accuracy of +/- 10%, which they say needs to be improved. Also, measuring distance to validate network traffic requires the use of radio equipment during the distance ranging which prevents it from being used for other needs.

To protect positional information, Time Difference of Arrival (TDOA) and Multilateration (MLAT) are proposed as methods to prevent injection of false data (Galati, G.; Leonardi, M.; Paciucci, 2005). Thurber (2012) reports that the FAA plans to keep half the current network of radar as a backup to ADS-B because of the risk of spoofing and jamming. Multileteration is a technique that uses the time difference between ADS-B signals arriving at different base stations to determine the position of the signal source (Linares, Chaves-Jacob, Julien, Schwenke, Longstaff, Andrew, Fletcher, Simon & Flore, Jakob & Uhlmann, Eckart

& Wintering, and Jens, 2014). Multilateration is limited, however, in that it will only work on aircraft-to-ground and not air-to-air signals (Thurber, 2012). There is also the risk that profession cyber hackers can recon the fixed ground stations and analyze their signals to overcome multilateraion (Thurber, 2012).

Another proposal with promise is array signal processing. Array Signal Processing is analysis done using a collection of antennas. The purpose of the analysis is to extract data about the incoming signals such as position. Naganawa, Chomel, Koga, Miyazaki, & Kakubari (2017) assert that anti-jamming and anti-spoofing must be provided together and conducted experiments on the effectiveness of array signal processing. Beamforming is a technique for transmitting or receiving radio waves preferentially in some directions compared with others.

Naganawa, et al found that beamforming antenna arrays can mitigate jamming, with the signal-to-jamming-plus-noise ratio being above the threshold for successful decoding of the signal. They further found that ADS-B locations can be verified with the angles of arrival of the ADS-B signals. Beamforming arrays are used to minimize interference from jamming by focusing on the signal of interest and minimizing radio frequency energy from the direction of interfering signals (Becker, 2014).

Some of the drawbacks to this approach include the cost of the sophisticated antenna arrays, which may cost millions of dollars. Also, many of the optimization algorithms assume that a signal comes from a fixed direction (Becker, 2014). At the same time, the communications channel can change over time because of weather or other external conditions change. A beamforming optimization algorithm must be able to quickly adapt to both these potential changes.

Literature on Detecting ADS-B Spoofing

The purpose of ADS-B spoofing is to disrupt airspace surveillance or to hijack airplanes or drones. With the advent of low-cost drones, the real risk in future is not a hacker on the ground using gnuRadio but the use of drones that have real flight paths (Kožović & Durdevic, 2021).

Spoofing poses more risk to the system than jamming because the targeted system, unaware it is under attack, cannot raise an alert to the rest of air traffic control. A replay attack is a form of ADS-B spoofing where authentic ADS-B messages are recorded and retransmitted later (Kožović & Durdevic, 2021). Different approaches to detecting spoofing have been proposed. These include cryptographic methods, signal separation, PVT (Position, Velocity, Time) verification, and Doppler Shift (Kožović & Durdevic, 2021).

Various organizations are monitoring the production of potential anti-spoofing detection methods. These include the ICAO (International Civil Aviation Organization), RTCA (Radio Technical Commission for Aeronautics) and EUROCAE (European Organization for Civil Aviation Equipment). Some of the methods proposed by these bodies include the use of alternative navigation equipment to mitigate the spoofing of GNSS and its impact on ADS-B.

Ying, Mazer, Bernieri, Conti, Bushnell and Poovendran (2019) describe three types of attacks: 1. Ghost aircraft injection; 2. Aircraft spoofing; and 3. IQ data replay (IQ or “in-phase” and “quadrature” are two signals which have the same frequency but differ in phase by 90°). Some attacks attempt to introduce a ghost plane into the air traffic control space. This can be from a ground-based Software defined radio (SDR) using either a replay attack in which previously recorded messages are inserted into the airspace or from a new message that introduces a ghost plane. Another type of attack is from the air, aircraft spoofing, which uses an ADS-B transponder with a modified ICAO-ID, the unique transponder identifier assigned by the ICAO.

Wang, Zou, and Ding (2020) propose a data mining approach to detect an ADS-B spoofing attack. Valid ADS-B messages are used to train a long short-term memory (LTSM) neural network, which is then used to discover irregularities in message sequences. Their testing involved inserting new messages with changed velocity and position, rather than a replay attack. Their model then detects that those changes are not legitimate. They form the ADS-B data into a sliding window, one of the input formats that can be processed by an LTSM neural net. The disadvantage of such an approach is the risk of over-fitting the data (Vishnu, Janik, Rezmer & Leonowicz, 2020). In addition, a replay attack will have valid position and velocity information.

Contribution of this Research

Most of the detection approaches for ADS-B spoofing in the literature review were based on electronic methods. In contrast, our approach is data driven. Two of the methods in the review were also data-driven approaches. Wang, et al (2020) use a long short-term memory model that is trained on valid ADS-B data downloaded from a GitHub site conducting ADS-B analysis. They then simulate bad messages to test their model. No replay tests were conducted, and their model, being trained with good ADS-B messages, may have difficulty recognizing replay messages as anomalous, because replay messages are reused good messages.

Ying, et al (2019) use a neural network classifier to detect replay attacks for recorded and then later retransmitted ADS-B messages. The model is trained to detect the differences in the IQ samples from the original aircraft transponder they recorded and those replay broadcasts from their ground station. Neither of these data-driven approaches tested their models against a real-time feed of ADS-B traffic to verify if the model could perform detection in real time. The contribution of this research is using a classic approach, cosine similarity, to do real time detection using message data.

Cosine similarity is a measurement calculation to tell us how similar two datasets are to each other. If the calculation is 1 then the two datasets are identical, while at 0 they have nothing in common. It is widely used to compare two different documents to gauge the similarity of the documents (Russell, 2013); in our case, the single 56-bit hexadecimal message (essentially a 14-character word, four bits per hex character) is our word and the set of messages for an ICAO-ID is a document. We will get incoming messages for every ICAO-ID currently in traffic space and we want to compare the set of messages from an incoming transponder with what is in store. We collect the message-set from the store for any incoming ICAO-ID and compare that with the incoming message-set. If a cosine is above a threshold of similarity, then we flag the culprit incoming ICAO-ID message-set as a possible replay attack.

Methodology for this research

Experimental Design

A replay attack inserts previously recorded ADS-B messages into the airspace introducing false messages to air traffic control. This research uses a data driven detection approach, cosine similarity, that is detailed in Step 4 and Step 5 subsections below. In developing the methodology for conducting a replay attack, several considerations come into play. Based on the literature review certain design aspects are clear, which are discussed in this section. An analysis of actual ADS-B data revealed other design considerations.

Not every message needs a cosine calculated but, when needed, the calculation was slightly over 200 milliseconds in our initial solution. The design uses parallel processing for scalability. The area of operations is the northwest region of Dulles International Airport, which also includes several regional airports and the 167th Airlift Wing of the WV Air National Guard. MRB Aviation in Martinsburg is also a reliever airport for the Baltimore-Washington corridor.

The authors' laptop computer had 8 logical processors and the design is able to spread calculation load over multiple processors. Using shared memory parallelism, the detection system should be scalable and able to keep up with the incoming data stream with increasing loads. The approach to parallelism is shown in Figure 2.

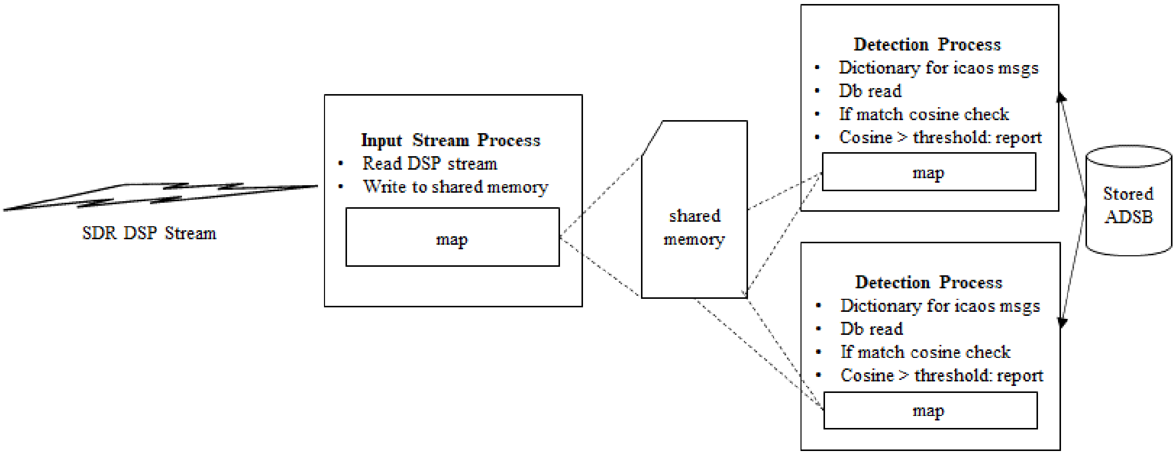

Figure 2: Detection Program Context Diagram

There is no timestamp information in ADS-B messages (Stroman, 2021; Sun, 2021), so that cannot be used for detecting a replay attack. Another important consideration is the Mode S downlink format to use. As noted above, this research is dealing with Mode S Extended Squitter, 1090ES, with three Mode S downlink formats that can be considered: 1.) DF 17 for ADS-B messages coming from a transponder that can be interrogated by the ATC system; 2.) DF 18 for ADS-B messages coming from a system that cannot be interrogated; and 3.) DF 19 for military avionics (Sun, 2021; FAA, 2016). The other downlink formats on 1090 MHz are various Mode S interrogate/reply interactions between the ATC and individual aircraft transponders. Detection in this research focused on DF 17 and 18 messages.

Our approach is to compare messages coming from an ADS-B transponder with a database of saved traffic to find an indication that the spatial information is identical to one broadcast in the past, a replay attack. The next sections will explain the methodology to integrate an ADS-B system with a microcomputer to do detailed analysis. Also explained is the structure of the database, loading ADS-B data into the database and the algorithm for determining if an input stream of live messages represents a replay attack. Systematically, the methodology follows the steps indicated in Table 3:

Table 3: Methodology for Detecting ADS-B Replay Attacks

Step 1	Configure hardware and software to stream 1090 MHz signals as data
Step 2	Read raw data from radio transponder into a data file
Step 3	Clean up raw data and load cleaned data into relational database
Step 4	Analyze data to understand context and limitations of attacks
Step 5	Develop the method for determining if an input stream contains a replay attack

Step1: RTL-SDR configuration for ADS-B 1090ES

Until recently, radio signals were decoded and processed by analog electronic circuitry. As microcomputers became more powerful, the mathematics to process radio signals could be performed quickly enough by software. In addition, Linux engineers developed drivers that transformed the RTL2832U chip into a wideband, general purpose radio receiver (Laufer, 2014). The methodology in this paper uses an RTL2832U based dongle that attaches to a microcomputer via the USB interface. A coax cable connects the receiver to a single pole antenna. Software drivers are installed that enable the operating system to communicate with the radio receiver as shown in Figure3. ADS-B messages are transmitted at 1090 MHz.

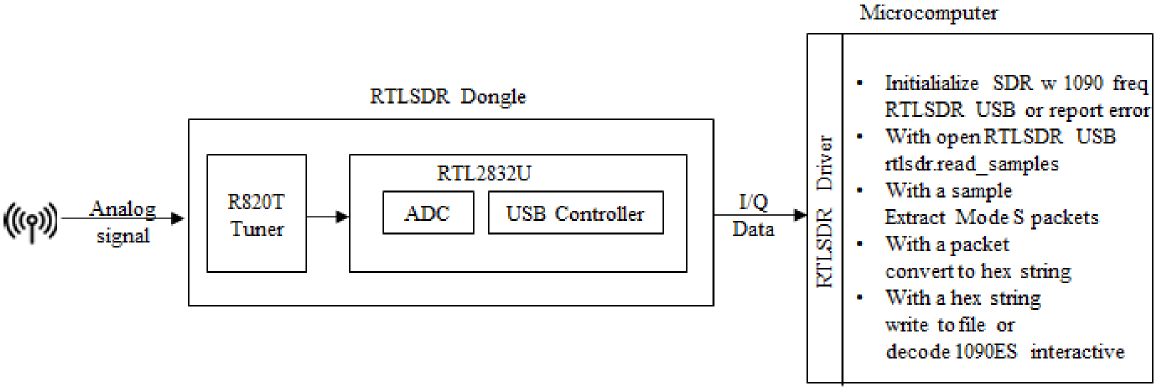
Microcomputer
RTLSDR Dongle
Analog signal
R820T Tuner
RTL2832U
ADC
USB Controller
I/Q Data
RTLSDR Driver
Initialialize SDR w 1090 freq RTLSDR USB or report error
With open RTLSDR USB rtlsdr.read_samples
With a sample Extract Mode S packets
With a packet convert to hex string
With a hex string write to file or decode 1090ES interactive

Figure 3: Software Defined Radio as used in this research.

An analog-to-digital converter on the RTL2832U samples the radio waves and digitizes the samples (Laufer, 2014; Lichtman, 2023). The resulting I/Q data, In-Phase and Quadrature, is made available by the USB controller on the RTL2832. I/Q is a precise representation of a signal (Kuisma, 2021). The software digital signal processing on the microcomputer can then process the I/Q data as appropriate. Either the I/Q data or the processed data of a signal can be used in detection. The approach used with the RTL2832U provides the authors with either I/Q or hexadecimal messages processed from the I/Q data. We used processed hexadecimal messages.

Step 2: Read raw data from radio transponder into a data file.

A program accessed the RTL-SDR driver to read the 1090MHz radio signal. The Mode S data streamed from the transponder has the following look and it can be further decoded using the data structure templates noted in previous sections of this paper. This becomes the stored data that is searched to determine if an incoming message was already broadcast sometime in the past.

```
02E19718E06942
5DC07995204451
8DACBF63990D769330049B495802
02E60F9EAA00CB
…
```

The ADS-B message transmissions use DF 17 and 18, of the different Mode S downlink formats (Sun, 2021). The other downlink formats are for various Mode S interrogate/reply messages not directly related to ADS-B (Wolff, 2012). Over 42,808,866 Mode S messages were collected, filtered, cleaned and transformed.

Step 3. Clean up raw data and load cleaned data into relational database.

This raw feed was filtered to select only DF 17 and 18 messages, the civilian ADS-B traffic. There were 14,414,774 ADS-B messages loaded into a relational database. The following database fields were stored in a table using the SQLite3 relational database system:

- Surrogate primary key,
- Downlink format and capability,
- ICAO-ID,
- Extended squitter message,
- CRC check, and
- Raw feed.

The raw ADS-B data was stored intact and parsed into the major fields of ADS-B data. The data was stored with redundancy to avoid costly joins of the data when processing large numbers of records.

Step 4. Analyze data to understand context and limitations of attacks.

The methodology for Step 5, detecting a live replay attack, is dependent on what the analysis of the ADS-B traffic data in Step 4 yields. A simple approach for detection would be to index the ADS-B table, which contains the messages that are stored and could be used for replay, and lookup the incoming messages for a match. This will not work, however, if there are naturally occurring duplicates because the match may be the result of a naturally occurring duplicate rather than a replay. So, the first analytic query is to find out if there are duplicate messages in the stored data. We searched for duplicates within an ICAO-ID message-set and between message-sets of different ICAO-IDs. A SQL query was formulated that searched for how many duplicates, if any, are in the stored message table.

Naturally occurring duplicates were discovered in the analysis. With duplicate messages naturally occurring both within and between ICAO-IDs there will be database matches with incoming traffic even without a replay attack. To find the threshold resulting from legitimate duplicates, the 13,481 ICAO-IDs were compared (cosine calculated) with each other. Once an ICAO-ID's message had been compared with cosine similarity to all the other ICAO-IDs in the database, it was removed from the temp table used for the determination so its results would not be double counted. As a result, n*(n+1)/2 or 90,875,421cosine similarity comparisons were conducted.

Findings from data analysis

Of the 14,414,774 messages 923,088 were found to have one or more duplicates. In reviewing the duplicates, many of them were found to have the same ICAO-ID – in other words came from the same aircraft in the short timeframe it was within range of the RTL-SDR. Likewise, many were found to be in the message-sets of different ICAO-IDs. With the possibility of naturally occurring duplicates, a database simple lookup by message will not suffice. A set of incoming messages from an aircraft needs to be matched with sets of stored messages to determine how many messages they share.

To match sets of messages to determine if they are similar, cosine similarity was chosen with the messages in a set becoming elements in a vector, each message a word and each message-set a document. The simple lookup is still used to indicate that a replay attack is possible, and a hit triggers the processing for cosine similarity. A last analysis was needed to determine the threshold, i.e., the cosine between sets that separates a replay attack from naturally occurring duplicates.

Once we exclude the .99 and 1.0 cosines that are an ICAO-ID compared to itself (similar to a replay attack), 5,606,912 of the 90,875,421 combinations had a cosine greater than 0. Figure 4 is a distribution of cosines calculated for all combinations of message-sets in the data. Data point one, on the far left of the chart in Figure 4, had a frequency count of 4,839,144, for a cosine of .01 and less, and a cumulative percentage of total at 86.3% for that cosine bin. We found that 99.999% of the calculated cosine scores between ICAO-ID vectors sharing duplicate messages were below a cosine of .07. This was selected as the threshold for identifying a replay attack when analyzing incoming messages. Two sets of messages had a cosine of .288, a percentage of 0.000000537%.

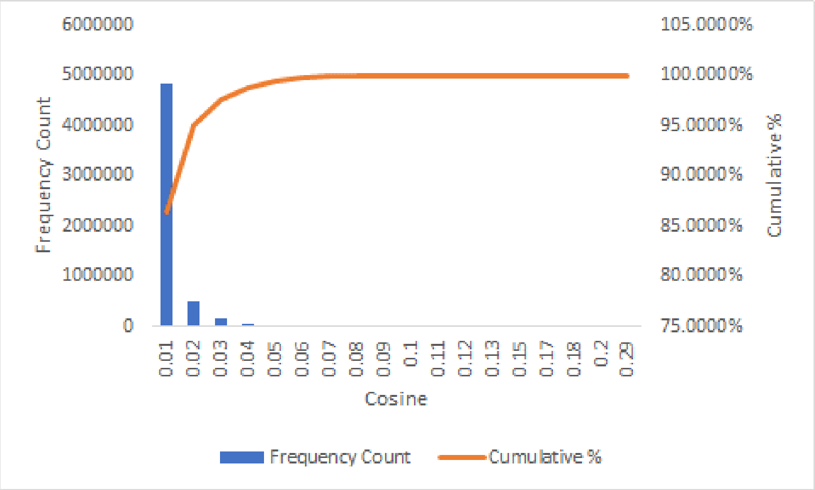

Figure 4: Binned Cosine Frequency Distribution

Step 5: Cosine Similarity Detection

The need is to compare two large sets of hexadecimal messages to rapidly find out the extent they match up. The first set is the incoming ADS-B messages for an ICAO-ID sent by the radio transponder. The second set is the stored messages for an ICAO-ID that shares at least one duplicate with the incoming message-set. The methodology for this step is to put each set into its own vector and do a similarity measure on those vectors. Cosine similarity is angle between the vectors, defined as the dot product of the vectors divided by the product of their lengths (Grus, 2015) and it was chosen for this research.

The research process is to take the 56-bit (14 hexadecimal character) incoming extended squitter messages from the 112-bit Mode S messages and do a search of each against the stored messages. If there is a match then the program loops through the database side first, collecting all the squitter messages corresponding to an ICAO-ID in the match. All the messages for an ICAO-ID are vectorized, ready for comparison with the incoming traffic. If there is more than one ICAO-ID in the match, each is vectorized.

Likewise, all extended squitter messages for the ICAO-ID of the incoming radio traffic are collected into a string that is vectorized. A cosine similarity is done on this vector and the vector created on the database side. If there is more than one vector on the database side, each is done in turn against the incoming vector.

If the cosine similarity is greater than 0.07 then the incoming ICAO-ID is flagged as a replay attack. The threshold of 0.07 was determined by an exhaustive analysis of the data as described above. The detection system must separate naturally occurring dups from an actual replay attack. Furthermore, it must be able to do this in real time. A series of tests were conducted to determine if the proposed system is capable of real time detection. Different configurations were performance tested to determine if any could keep up with the feed from the ADS-B transponder. These include single process, multithreaded, in memory database or in memory dictionary structures, key value pairs, native to the programming language. Each approach was timed.

In addition to these tests, the following tests were conducted to validate the detection of replay attacks.

- Single attack: a single ICAO-ID's message-set is fed into the message stream.
- Swarm attack: Multiple ICAO-IDs' message-sets are fed in at the same time.
- Spaced attack: Multiple ICAO-IDs; sets are fed in at different times.

Findings from Cosine Detection

Since it is against U. S. law to broadcast a replay attack over 1090MHz, we chose a data driven approach instead. As the incoming stream of radio traffic on 1090MHz is digitized, we interspersed a stored message-set and fed the resulting digital data stream of valid messages with a known replay into the detection program. In the initial configuration, the detection process could not keep pace with the incoming traffic.

Preliminary tests were done on a prototype of the application to determine which configurations provided performance improvements. A multithreaded version of the application was prepared but proved to add no value to performance. In fact, it consistently ran slower than the base app which did not use multithreading.

There were no indications of CPU or I/O pressure in the application, based on Performance Monitor (Perfmon) indicators. To discover the lag in the detection processing in relation to the feed of transponder messages, timers were put in the code. Database calls in the program resulted in significant delays – up to 200 milliseconds. Another preliminary test was run to determine if an in-memory database would improve performance. The database for one weeks' worth of messages is small, approximately 1.2GB. A comparison of the times with database on disc and database in-memory, likewise, found there was no improvement in performance with an in-memory database.

Our explanation for these outcomes is that the context switch when calling the database resulted in the delay. To test this, dictionaries, key-value data structures native to the programming language, were preloaded with the data from the database and used in the detection algorithm, which then made no outside calls to a database system. With this change, the detection program was able to provide real time processing of the incoming message feed.

Results

Different threshold levels were tried to determine the effect on real time performance. There were no appreciable differences in performance. At the extremes, in a test with a .07 cosine threshold, the message feed finished serving 40,000 messages in 12 minutes and 51 seconds while the detection processing finished the same 40,000 message-set in 12 minutes and 52 seconds. With a .70 cosine, message feed finished in 16 minutes and 13 seconds while the detection processor finished in 16 minutes and 18 seconds. The time difference between the .07 and .70 tests were the result of different periods of the day having different numbers of aircraft and with fewer aircraft, the flow rate of message traffic is less. The detection program performed cosine similarity on various replay attack scenarios. In validation testing, it was able to detect all the invalid ICAO-ID for all three tests: single, swarm and spaced. The tests were run on a laptop computer with 32 GB of RAM and 4 cores with 2 threads per core yielding 8 logical processors. Multiple processors ran in parallel, and this did not stress the CPU, memory or disk capacity.

Discussion

Scalability

A principal issue in using any data driven approach is the amount of data that needs to be stored and processed. This study focused on reading ADS-B messages transformed from a radio transponder along with stored data to detect replay attacks. As to scalability, our findings show that a weeks' worth of data can be readily handled by a single logical processor and 2.5GB of memory. Current servers can provide 896 logical processors and 6 Terabytes of memory. The needs for this detection system are: 1x52x10 = 520 processors and 2.5GB x 52 x 10 = 1300GB or 1.3TB of memory to handle a decade's worth of data. Our parallel processing architecture was tested with multiple processes running smoothly while working on the load from the transponder dongle. The replay detection system will have a geographically local scope. Latitude and longitude of the traffic space would be used to filter out replays of ADS-B messages from another location. One of the advantages of a replay attack is that the actual position is legitimate and may not be flagged for certain irregularities by neural network models.

Mitigations

Mitigations must also be considered. Detecting the presence of a replay attack is the first step in mitigating it. The GAO (2020) notes that once ADS-B spoofing is detected, air traffic controllers can begin actions to work with the systems involved to resolve the situation through controller pilot communications. In addition, the ICAO-ID of the replay attack could be sent to the surveillance radars that will still be present in NextGen for triangulation to fix the location of the culprit. It is also possible to change the icon in air traffic control displays for aircraft that are part of a replay attack to highlight this to the other elements in the system.

Conclusion

The approach taken in this study was able to detect ADS-B replay attacks. It used digital signal processing on a microcomputer to process the I/Q data made available through a USB controller into hexadecimal messages. A series of steps was involved in filtering out the actual ADS-B content and detecting a threshold of duplicates that warranted scoring data as a replay attack. The presence of naturally occurring duplicates prevented a simple lookup of a matching message and resulted in the need to vectorize suspect messages for similarity comparison.

Based on the findings, a key implication for air traffic management is that a decade's worth of ADS-B data and detection processing can fit on servers that are currently available. Scalability is the next topic for research and the authors are confident that modern servers are up to this task. The NextGen FMDS is scheduled for full implementation by 2030 and is required to have capabilities for handling large data stores (FAA, 2023). Another key implication is that replay attacks can be detected real-time and since the approach does not involve modification of the existing ADS-B protocol, it is a practical mitigation to replay attack vulnerabilities in the National Airspace System.

Cybersecurity is a competition with sentient actors. All detections and mitigations described in this paper are on-going efforts with new defenses required to meet new attacks. One lesson learned is that anti-malice methodology is better introduced in the protocol design phase rather than in application software development. This paper inventoried numerous other exploitation opportunities. In addition to extending our research in terms of scalability, our other future directions are to look at a data driven approach for GNSS spoofing attacks, as well as detecting invalid calculations from the WAAS.

References

Alghamdi, F., Alshrahani, A. & Hamza, N. 2018. Effective security techniques for automatic dependent surveillance-broadcast (ADS-B). *International Journal of Computer Applications*, 180(26), pp.23-28 [online]. Available at: https://www.ijcaonline.org/archives/volume180/number26/alghamdi-2018-ijca-916598.pdf

Becker, J. (05/15/2014). Dynamic Beamforming Optimization for Anti - Jamming and Hardware Fault Recovery. Kilthub Carnegie Melon University

Burfeind, Brandon C., "Interoperable ADS-B Confidentiality" (2020). Theses and Dissertations. 3156. https://scholar.afit.edu/etd/3156

Elofson, S., B. Redondo, M. Francetic, M. Nelson, M. White, and M. Winter (2018). STRAIGHT TALK ABOUT ADS-B. Duncan Aviation. Retrieved from https://www.duncanaviation.aero/files/straight-talk/Straight_Talk-ADS-B.pdf

FAA (2016). Non-Transponder-Based Automatic Dependent Services -Broadcast (ADS-B) Downlink Format 18 (DF=18) Virtual Target Injection. Federal Aviation Administration William J. Hughes Technical Center. DOT/FAA/TC-16/5

FAA (2023). Forming NextGen: From Vision to Reality. FAA. Retrieved from https://www.faa.gov/nextgen/background/forming

Galati, G.; Leonardi, M.; Paciucci, V. (2005). Wide area surveillance using SSR mode S multilateration: Advantages and limitations. In Proceedings of the 2nd European Radar Conference EURAD, Paris, France, 3–8 October 2005; pp. 225–229.

George, M., S. Tuladhar & S. Sivananthan (2020). A study of uncompensated latency in ADS-B reports. Arcon Corporation. Retrieved from https://www.researchgate.net/publication/343689163_A_study_of_uncompensated_latency_in_ADS-B_reports

Government Accountability Office. (January 18, 2018). Homeland Defense: Urgent Need for DOD and FAA to Address Risks and Improve Planning for Technology That Tracks Military Aircraft. (GAO Publication No. 18-177). Washington, D.C.: U.S. Government Printing Office.

Government Accountability Office. (October 2020). AVIATION CYBERSECURITY. FAA Should Fully Implement Key Practices to Strengthen Its Oversight of Avionics Risks. (GAO Publication No. 21-86). Washington, D.C.: U.S. Government Printing Office.

Goward, D. NASA Report: Passenger Aircraft Nearly Crashes Due to GPS Disruption, GPS World. Available online: https://www.gpsworld.com/nasa-report-passenger-aircraft-nearly-crashes-due-gps-disruption (accessed on 24 December 2022).

Grus, J. (2015). Data Science from Scratch. O'Reilly.

Kožović, D. & D. Đurđević (2021). SPOOFING IN AVIATION: SECURITY THREATS ON GPS AND ADS-B SYSTEMS. VOJNOTEHNIČKI GLASNIK / MILITARY TECHNICAL COURIER, 2021, Vol. 69, Issue 2

Kuisma, M. (11/26/2021). I/Q Data Explained. Retrieved from https://www.pe0sat.vgnet.nl/sdr/iq-data-explained/ and http://whiteboard.ping.se/SDR/IQ

Leonardi, M.; Sirbu, G. (2021). ADS-B Crowd-Sensor Network and Two-Step Kalman Filter for GNSS and ADS-B Cyber-Attack Detection. Sensors 2021, 21, 4992. https://doi.org/10.3390/s21154992

Laufer, C. (May 14, 2014). The Hobbyist's Guide to the RTL-SDR 7e. RTL-SDR.com.

Lichtman, M (2023). PySDR: A guide to SDR and DSP using Python. Retrieved from https://pysdr.org/index.html

Linares, Jean-marc & Chaves-Jacob, Julien & Schwenke, H. & Longstaff, Andrew & Fletcher, Simon & Flore, Jakob & Uhlmann, Eckart & Wintering, Jens. (2014). Impact of Measurement Procedure when Error mapping and Compensating a Small CNC Machine using a Multilateration Laser Interferometer. Precision Engineering. 38. 578-588. 10.1016/j.precisioneng.2014.02.008.

Lux, J. & M. Schaefer (March 1, 2005). Displacing Unpredictable Nulls in Antenna Radiation Patterns. NASA Jet Propulsion Lab.

McCallie, D., J. Butts & R. Mills (7 July 2011). Security analysis of the ADS-B implementation in the next generation air transportation system. International Journal of Critical Infrastructure Protection, 4 (2011) p78–87

Naganawa, J., C. Chomel, T. Koga, H. Miyazaki, & Y. Kakubari. (Nov 2017). Jamming and Spoofing Protection for ADS-B Mode S Receiver Through Array Signal Processing. EIWAC 2017, 5th ENRI international workshop on ATM/CNS, Nov 2017, Tokyo, Japan. pp.184-204. ⟨hal-02191062⟩

Nelson, M. (2016). STRAIGHT TALK ABOUT WAAS/LPV. Duncan Aviation. Retrieved from www.DuncanAviation.aero/straighttalk

Rawat, R., Chakrawarti, R. K., Vyas, P., Gonzáles, J. L. A., Sikarwar, R., & Bhardwaj, R. (2023). Intelligent Fog Computing Surveillance System for Crime and Vulnerability Identification and Tracing. International Journal of Information Security and Privacy (IJISP), 17(1), 1-25.

Ronen, R. & B. Ben-Moshe (2021). Cyberattack on Flight Safety: Detection and Mitigation using LoRa. *Sensors 2021,21,4610*.

Russel, M. (2013). Mining the Social Web. O'Reilly.

Stroman, S., "Automatic Dependent Surveillance Broadcast (ADS-B) Security Mitigation through Multilateration" (2021). UNF Graduate Theses and Dissertations. 1106. https://digitalcommons.unf.edu/etd/1106

Sun, J. (2021). The 1090 Megahertz Riddle. Retrieved from https://mode-s.org/decode/content/ads-b/1-basics.html

Thurber, M. (September 3, 2012). ADS-B Is Insecure and Easily Spoofed, Say Hackers. *AINonline* retrieved from https://www.ainonline.com/aviation-news/aviation-international-news/2012-09-03/ads-b-insecure-and-easily-spoofed-say-hackers.

Vishnu, S., P. Janik, J. Rezmer & Z. Leonowicz (Feb 2020). Forecasting Solar PV Output Using Convolutional Neural Networks with a Sliding Window Algorithm. Energies, Vol. 13 Issue 3, p723-723.

Wang, J., Y. Zou, and J. Ding (2020). ADS-B spoofing attack detection method based on LTSM. EURASIP Journal on Wireless Communications and Networking, 2020:160.

Wolff, C. (3/4/2012). Mode S Reply Encoding. Retrieved from https://www.radartutorial.eu/druck/index.html and https://www.radartutorial.eu/13.ssr/sr24.en.html#:~:text=Each%20Mode%20S%20downlink%20format,for%20the%20ADS%2DB%20system

Ying, X., Mazer, J., Bernieri, G., Conti, M., Bushnell, L. & Poovendran, R. (2019). Detecting ADS-B spoofing attacks using deep neural networks. In: *IEEE Conference on Communications and Network Security (CNS)*, Washington DC, USA, June 10-12. Available at: https://doi.org/10.1109/CNS.2019.8802732