Journal of Open Aviation Science (2023), Vol.**1**
doi:10.59490/joas.2023.7205

JOAS

PROCEEDINGS | ***11th OpenSky Symposium***

Scaling the Timing-Based Detection of Anomalies in Real-World Aircraft Trajectories

Lukas Baege,[1] Patrick Schaller,[1] Vincent Lenders,[2,3] and Martin Strohmeier[*,2,3]

[1]ETH Zurich, Zurich, Switzerland
[2]Cyber-Defence Campus, Zurich, Switzerland
[3]OpenSky Network, Burgdorf, Switzerland
[*]Corresponding author: martin.strohmeier@armasuisse.ch

(Received: 24 October 2023; Revised: 7 February 2024; Accepted: 28 March 2024; Published: 1 April 2024)
(Editor: Tatiana Polishchuk; Reviewers: Vishwanath Bulusu, Andrei Gurtov, and David Lovell)

Abstract
The insecurity of ADS-B and GPS is a well-known problem in air traffic control. In this work, we introduce a statistical location verification method for crowdsourced aircraft data. We implement this method on the backend of the OpenSky Network and evaluate it on 8 months of data collected from Radarcape sensors. We thus demonstrate the effectiveness of our method in finding anomalous flight data in faster than real time. While we do not find clear evidence of widespread malicious spoofing attacks captured by OpenSky sensors, many sensor and transponder issues are unearthed. Finally, we simulate ADS-B and GNSS attacks, which were detected successfully with few false positives.

Keywords: ADS-B security; location verification; multilateration

1. Introduction

Automatic Dependent Surveillance-Broadcast (ADS-B) is a technology that allows aircraft to broadcast their location, speed, altitude, and other information to air traffic control (ATC) stations and to other aircraft without requiring human action. ADS-B is intended to improve safety, efficiency, and situational awareness in aviation by providing the involved parties with a real-time overview of the surrounding airspace. These advantages have led to the widespread adoption of ADS-B as a regulatory standard for safeguarding air travel across the globe. [1]

However, despite its advantages, researchers have been voicing concerns about the security aspects of ADS-B for over fifteen years now [2], as ADS-B messages are neither encrypted nor authenticated. The lack of *encryption* has positive implications, such as maximized compatibility and interoperability. It also enables facilitates crowdsourced ADS-B information-sharing networks supported by hobbyists, making flight data available to the public. The lack of *authentication*, however, has serious consequences. An adversary with the capability to send radio signals in the 1090 MHz frequency range can craft and transmit valid ADS-B messages. As a consequence, adversaries within the radio signal range of an ATC station can inject ghost aircraft or modify flight tracks of legitimate aircraft in ATC systems [3] and aircraft [4] using a cheap software-defined radio (SDR).

Another security issue that has received much attention is the reliance of aircraft on insecure global navigation satellite systems (GNSS). Most prominently, the global positioning system (GPS) does not

implement any mechanisms to guarantee authenticity in its civilian-use version, giving adversaries the ability to spoof satellite signals and advertise arbitrary GPS data. If sufficiently strong spoofed signals are directed toward an aircraft, its navigation systems may be deceived, causing the aircraft to deviate from its intended course and transmit incorrect location data over ADS-B. [**gnssoofing**]

Various measures of verifying ADS-B location claims have been proposed in the literature. If only a single sensor is available, plausibility checking can detect nonsensical location, heading and velocity claims in flight paths, and physical-layer features like the signal's direction of arrival can be used to detect impossible location claims. With multiple receivers, cross-checks can be made based on the received signal strength, time-of-arrival data, or physical-layer properties of the signal. With four or more sensors available, the transmitter location can be reconstructed using multilateration [5].

Previous work evaluated location verification using ADS-B sensor network data [6, 7, 8, 9] but typically assumed the data to be clean, and focused on detecting simulated, manually injected, attacks.

In this paper, we use real data from the OpenSky Network [10], a crowdsourced ADS-B sensor network, to detect inconsistencies in aircraft's location claims using statistical analysis of Time-Difference-of-Arrival (TDoA) measurements at scale. Using crowdsourced data for securely verifying flight tracks presents interesting challenges. A key differentiation to most previous research is that we do not fully trust the sensor data, as an attacker could easily register own sensors to the network, and inject manipulated records. Additionally, even non-malicious sensors can report misleading or erroneous data. Evaluating which sensors provide precise and accurate enough data for TDoA-based location verification is in itself an interesting challenge, also since the sensor selection process has to be fully automated due to the dynamic topology of a crowdsourced network.

We use our system to analyze 8 months of air traffic data and investigate the encountered inconsistencies in the data. On average, over 2 billion sensor records were analyzed for each day, with our data set totaling 83 Terabytes of data.

1.1 Contributions

- We develop a threat model suitable for a crowdsourced sensor network.
- We develop methods to statistically verify TDoA measurements of ADS-B position messages under the assumed threat model.
- We perform exploratory analysis of a large real-world data set and analyze the detected issues with sensor and aircraft equipment.
- We show that our system can reliably detect certain attack types and non-malicious inconsistencies.

2. Background

2.1 The OpenSky Network

The OpenSky Network is a crowdsourced sensor network, collecting air traffic data in the form of ADS-B messages from aircraft transponders. It allows users to register their own sensors to the network and displays real-time air traffic data based on information provided by registered sensors. OpenSky provided the computational capacity and historical data needed for this research.

The data stored on the server contains the raw Mode S message along with information about the sensor, including a unique identifier and a claimed location and hardware type. Each record include a sensor-reported timestamp of when the signal has been received, as well as a server timestamp of the receipt of the message at the central OpenSky server over the internet.

2.2 Automatic Dependent Surveillance–Broadcast

ADS-B allows aircraft to automatically transmit information such as position, altitude, speed, and other relevant data on the 1090 Mhz channel to ground-based ATC systems and other aircraft. This allows ATC systems to monitor the position and trajectory of aircraft in real-time, improving the efficiency and safety of air traffic management. Fig. 1 provides the basic message structure. Notably, the protocol does not provide timing information.

Figure 1. A Mode S message carrying ADS-B data is 112 bits long. It encodes the downlink format (DF), transponder capabilities (CA), ICAO 24-bit aircraft identifier (ICAO), the extended squitter message (ME), and parity information (PI). The ME field contains the 5-bit ADS-B type code and a 51-bit data field.

ADS-B Security

Despite the advantages of ADS-B for modern air traffic management, overconfidence in the system may have dangerous consequences, as practical attacks on ADS-B receivers have been publicly demonstrated since 2012 [3]. These kinds of attacks are easy to perform as ADS-B messages lack security features that ensure confidentiality and authenticity of the broadcast messages.

2.3 Global Navigation Satellite Systems

Global navigation satellite systems provide satellite-based high-accuracy positioning with global coverage. The most prominent GNSS is GPS provided by the USA but other constellations exist, such as Galileo by the European Union, or GLONASS operated by Russia.

GNSS Security

As aircraft use GNSS to obtain their current position, any interference with GNSS presents another attack vector on ATC. Civilian GNSS-based localization systems do not usually provide any authentication. This allows an attacker to send fake GNSS signals to a target aircraft, which will use the spoofed signal to derive location information, provided that the attacker's signal at the target aircraft is stronger than the legitimate signals. While this attack is known since the early 2000s, and demonstrated e.g. on ships [11], GNSS spoofing attacks on commercial aircraft have been publicly confirmed in the field for the first time in 2023. [12, 13] GPS jamming, in contrast, has been a common occurrence not only in conflict zones for many years now [14].

3. Problem Constraints

3.1 System Model

- We assume that aircraft predominately use GNSS for navigation and can be effectively interfered with by GNSS spoofing as demonstrated in recent indents. [12]
- We restrict our analysis to ADS-B position messages.
- We assume flight tracks are generated from ADS-B messages similar to OpenSky's model [15].

Crowdsourced Sensor Networks

The core system is a crowdsourced network of ADS-B sensors. During the operation of the network, sensors can be added and removed at any time. The location of each participating sensor is assumed to be fixed and known, either provided by the user or by using independent positioning methods [16]. For each received ADS-B message, a sensor will forward the raw message bytes, an indicator

of the time of reception, and a unique sensor identification number to a central server. For each message, the collection server additionally stores the timestamp when the message was received.

We assume that the sensors' clocks remain accurate over time, meaning they exhibit no clock drift. This is an optimistic assumption [17] but the sensors used in our analysis generally showed negligible clock drift over relevant time scales. We do not require the clocks to be synchronized, though, allowing for constant offsets between sensors.

3.2 Threat Model

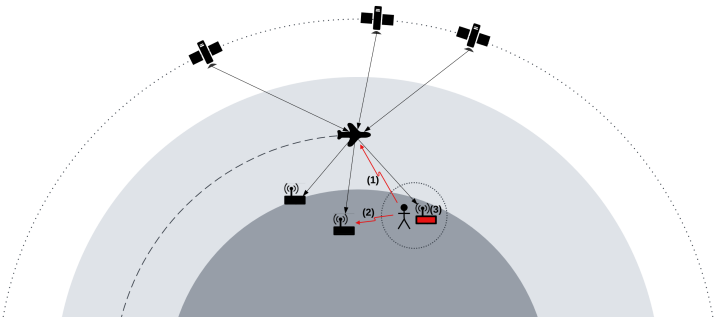

Figure 2. Threat Model. An adversary can (1) send spoofed GNSS signals to an aircraft, (2) send spoofed ADS-B messages to the sensor network, or (3) directly manipulate data coming from individual sensors

Our proposed approach is designed to detect incorrect ADS-B location data. This false information can be caused by an attacker broadcasting fake ADS-B messages, a GNSS spoofing attacker leading an aircraft off track, or it might simply be caused by broken or inaccurate equipment on the aircraft or the ADS-B receiver side. In our threat model, the attacker has (1) the ability to transmit spoofed GNSS signals to aircraft using directional antennas, (2) broadcast arbitrary ADS-B messages to one or more OpenSky sensors, and (3) to add malicious sensors to the OpenSky Network or compromise legitimate sensors (see Figure 2). The ability to perform an insider-attack is inherent to crowdsourced networks, and prevents us from ever fully trusting any single sensor, as a malicious sensor might try to gain the trust of an attack detection system before reporting misleading data, and legitimate sensors may get compromised over time. To retain the ability to detect false location advertisements, we set some limitations on the amount of fake and erroneous data received by the sensor network.

First, we limit an attacker to a single transmitter, disallowing multi-device attacks. This rules out very sophisticated attackers, which could, knowing the exact locations of the target sensors, send the same message using low-power signals in the vicinity of multiple sensors at pre-calculated timings to mimic the true TDoA data generated by a legitimate flight track. Attacks like these can be detected by performing verification tests of physical-layer characteristics at multiple receivers [18].

We further limit the number of false ADS-B flight tracks an attacker can inject into the network and how many sensors they can control using unified constraints. For this, we first define the term *sensor-pair witnessing* as follows: We say a sensor pair (s_i, s_j) witnessed a flight track f, if the sensors s_i and s_j have received some overlapping subset of location messages for flight track f. We assume for each legitimate sensor that more than half of all sensor-pair observations that include this sensor are legitimate, meaning the paired sensor as well as the received ADS-B data is legitimate. Additionally, for every flight track, legitimate or not, we require that less than half of the sensor-pair observations of this flight track include a malicious sensor.

These constraints limit the number of false flight tracks that can be broadcast by an adversary to less than half of all flight tracks in any geographical region.

3.3 Quality of the Input Data

One of the main challenges, when using TDoA measurements from crowdsourced sensors for the detection of ADS-B spoofing attacks, is the quality and reliability of the sensor data. Crowdsourced sensors are typically low cost and have varying levels of clock accuracy and precision, which makes automated data selection and filtering a challenge. In practice, we need to check each sensor and validate that it performs within our accuracy requirements.

There may be gaps in coverage where signals are not received by at least two sensors in the network. The use of a wireless medium for ADS-B transmissions also introduces multi-path effects between the sender and receivers. This can lead to inconsistent message reception timestamps on the receiver side, or even to messages being received multiple times at a single receiver.

4. Methods

The large scale of our analysis — many Terabytes of data — requires computationally efficient algorithms. To stay within memory limitations, and due to the way the data is structured on OpenSky's HDFS file system, we analyzed the data in one-hour batches.

The input data comes in the form of records, with information about the receiving sensor and the raw message bytes. To facilitate analysis of flight tracks, we group the records by the ICAO 24-bit aircraft identifier field of the ADS-B messages.

4.1 Preliminary Data Filtering

Our proposed methodology should work with all types of sensors, as we perform a sensor validation step which checks the consistency of all sensors' timestamps. However, in order to save computational resources, we narrow down our data set to only include records from sensors of type *Radarcape*, as these report more precise timestamps compared to self-made setups based on *dump1090.*.

Because Central Europe currently has the highest sensor density (see Figure 4), we further narrowed down our data to this general region.

Finally, we remove all ADS-B messages that do not encode location information, such as aircraft identification or status messages, and remove location messages received by only one sensor, as these cannot be analyzed using TDoA methods.

4.2 Estimating Characteristic Variance Scores

Our goal is to test whether the TDoA measurements of a given sensor pair follows the distribution which we would expect if we assume the aircraft's position reports and the two sensors' timing data are accurate. If the measurements do not match our expectation, we conclude that the advertised flight track or the data from at least one of the sensors was wrong. We use this information to disqualify bad sensors and to detect illegitimate flight tracks.

To determine the expected TDoA data from a flight track f and sensors s_i and s_j, we calculate the *expected difference* in signal arrival time between the two sensors $t_i - t_j$ for every ADS-B location message received by both sensors. We compute the difference as follows:

$$t_i - t_j = \frac{dist(loc_{source}, loc_i) - dist(loc_{source}, loc_j)}{c} \tag{1}$$

Here, $dist(.)$ refers to the Euclidean distance between two locations. To allow us to efficiently and accurately calculate these distances, we convert all geographical coordinates from the familiar World

Geodetic System, which uses latitude, longitude, and altitude to identify Earth-based locations, to the Earth-centered, Earth-fixed (ECEF) coordinate system. In the ECEF system, locations are described using 3-dimensional vectors originating at the Earth's center of mass.

For the measured timestamps, we expect the following sources to introduce certain deviations from the expected values, even for legitimate flight tracks and sensors:

- The aircraft's ADS-B location claims may be inaccurate (GPS malfunction or not existent).
- Multi-path effects may impact the signal's arrival time at the sensors.
- The sensors may measure the signal's time of arrival inaccurately.
- The true sensor locations may be slightly offset from their claimed locations.

The exact nature of the introduced errors is difficult to predict. We chose to model the overall error introduced by the listed sources as a random variable following a normal distribution. We assume that for a given flight track f and a sensor pair (s_i, s_j), the error of each ADS-B message's TDoA measurement is drawn from the same distribution, as the hardware-specific properties of the involved components, like the aircraft GNSS receiver and the sensor clocks, are expected to stay consistent over the time frame under analysis. We therefore denote our random variable as follows:

$$X_{f,i,j} \sim N(\mu_{f,i,j}, \sigma^2_{f,i,j}) \tag{2}$$

We call $\sigma^2_{f,i,j}$ the **characteristic variance of flight track f and sensor pair (s_i, s_j)**. Generally, this variance will be small for flight tracks in which the quality of the aircraft's location reports, and the sensor's timing and location information is accurate, and large otherwise. In the following section, we derive how these characteristic variances can be estimated.

We model all sensor clocks to exhibit no clock drift over the analyzed time frame, but do not require them to be synchronized with each other. This motivates the introduction of the following relation between a timestamp $\hat{t}_i$, as measured by sensor s_i in its internal time reference, and the corresponding timestamp t_i in our global time reference:

$$\hat{t}_i = t_i + \Delta_i \tag{3}$$

Due to the lack of clock drift in our system model, Δ_i remains constant. When we apply this offset to TDoA measurements, $\hat{t}_i$ and $\hat{t}_j$ represent the timestamps of the signal's arrival time, as measured by the sensors s_i and s_j respectively, in their internal time references.

$$\hat{t}_i - \hat{t}_j = (t_i + \Delta_i) - (t_j + \Delta_j) \tag{4}$$

$$\Rightarrow \hat{t}_i - \hat{t}_j = (t_i - t_j) + (\Delta_i - \Delta_j) \tag{5}$$

Because we model the TDoA measurements according to Equation 1 but with the addition of random variable $X_{f,i,j}$, which captures errors in time measurements by sensors s_i and s_j for messages related to flight track f, we get:

$$\Rightarrow \hat{t}_i - \hat{t}_j = \left(\frac{dist(loc_{source}, loc_i) - dist(loc_{source}, loc_j)}{c} + X_{f,i,j} \right) + (\Delta_i - \Delta_j) \tag{6}$$

All locations loc_* refer to the claimed, but not yet validated, locations of the signal source and the sensors. We rearrange the equation as follows:

$$\Rightarrow \hat{t}_i - \hat{t}_j - \frac{dist(loc_{source}, loc_i) - dist(loc_{source}, loc_j)}{c} = X_{f,i,j} + (\Delta_i - \Delta_j) \tag{7}$$

Now, the left-hand side consists of only known, measurable or derivable quantities: The measured timestamps $\hat{t}_i$ and $\hat{t}_j$ as well as all of the claimed locations are directly available from the data. The parameters of the normal distribution, as well as the constant offsets Δ_i and Δ_j, are unknown. Given the distribution for $X_{f,i,j}$ specified in Equation 2, we can consolidate the right-hand side of Equation 7 and express the left-hand side as a random variable:

$$X_{f,i,j} + (\Delta_i - \Delta_j) \sim N(\mu_{f,i,j} + \Delta_i - \Delta_j, \sigma^2_{f,i,j}) \tag{8}$$

$$\hat{t}_i - \hat{t}_j - \frac{dist(loc_{source}, loc_i) - dist(loc_{source}, loc_j)}{c} \sim N(\mu_{f,i,j} + \Delta_i - \Delta_j, \sigma^2_{f,i,j}) \tag{9}$$

Applying Equation 9 to our data set, we use the sample variance of the obtained values as an estimator for the characteristic variance $\sigma^2_{f,i,j}$.

For legitimate flight tracks f and well-behaved sensors s_i and s_j, we would expect their characteristic variance $\sigma^2_{f,i,j}$ to be small. If the observed variance is above a certain threshold, we conclude that either one of the sensors or the received ADS-B messages provide wrong information.

The distance between two sensors limits the maximum TDoA that can be observed by those sensors:

$$|t_i - t_j| \leq \frac{dist(loc_i, loc_j)}{c} \tag{10}$$

Because we allow the data set to contain measurement errors, for example, due to inaccurate sensor locations or timestamps, up to a given threshold, we only consider sensor pairs with sufficient distance between each other. This prevents the errors from dominating the measured time difference, while ensuring that each sensor pair can produce variance values above the thresholds used to flag problematic sensors and tracks.

4.3 Sensor Selection

The fact that we have no control over the sensors in the network and no detailed information about their hardware configuration or status, forces us to infer some of their properties to decide whether a sensor can be used for final flight track analysis. To do so, we first calculate all characteristic variance values for each flight track and pair of sensors as described in the previous section. In a next step, we define the set V_k to include all characteristic variance scores where sensor s_k is one of the involved receivers:

$$V_k := \{\sigma^2_{f,i,j} \in V | i = k \vee j = k\} \tag{11}$$

V denotes the set of all characteristic variances. Finally, we identify all sensors which exhibit low characteristic variance values and include those in the flight track analysis. In particular, a sensor s_k will be included if and only if:

$$med(V_k) \leq T_{sensor} \tag{12}$$

Here $med(.)$ denotes the median, and T_{sensor} is some appropriately chosen threshold value.

The use of the median for the selection of sensors is motivated by our threat model. If we assume that sensor s_k is well-behaved and less than half of the values in V_k are influenced by a spoofed flight track, or a malicious or misbehaving other sensor, the median value will stem from TDoA data as measured by s_k and another legitimate sensor, witnessing a legitimate flight track and therefore exhibiting a characteristic variance below the threshold. This will lead to the sensor s_k being utilized for our subsequent flight track analysis. If, on the other hand, the sensor s_k has poor clock precision, or other issues resulting in inaccurate sensor data, most of the values in V_k are expected to exceed the defined threshold, thereby excluding the sensor from final analysis. Note that malicious sensors can still be included, if they manipulate arrival timing information for only some flight tracks. This case is considered in the track verification step.

4.4 Track Verification

Once the sensors have been filtered, our final track verification will decide for each flight track whether the reported location data is consistent with the derived TDoA values, utilizing the same characteristic variance values that have already been used for sensor selection. For each flight track g, we check if the median characteristic variance of the flight track as observed by eligible sensor pairs is lower than some predefined threshold T_{track}:

$$med(\{\sigma^2_{g,i,j} \in V | s_i, s_j \text{ are considered good}\}) \leq T_{track} \tag{13}$$

If this check fails, we flag the flight track as problematic. Importantly, if a flight track passes this check, we cannot conclude that the reported flight path was accurate. Consider Figure 3 for examples of spoofing attacks which are undetectable in cases where not enough reliable sensors are available.

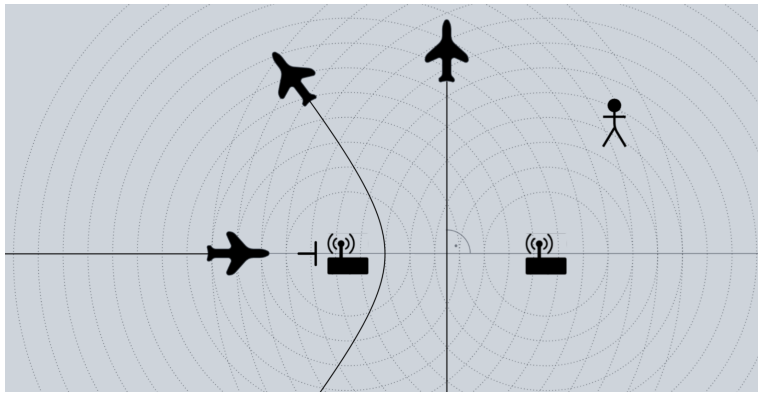

Figure 3. In this simplified 2-dimensional model, the shown flight tracks would all result in constant TDoA data, resembling a stationary attacker, as long as the aircraft coming from the left does not continue beyond the first sensor.

The use of the median here is again motivated by our threat model. Because we require that more than half of the sensor pairs witnessing the flight contain no illegitimate sensor, the median protects us from both high and low malicious variance data. Furthermore, the median serves as a robust measure against outlier data.

Generally, detection accuracy will improve with higher sensor density and more precise sensors.

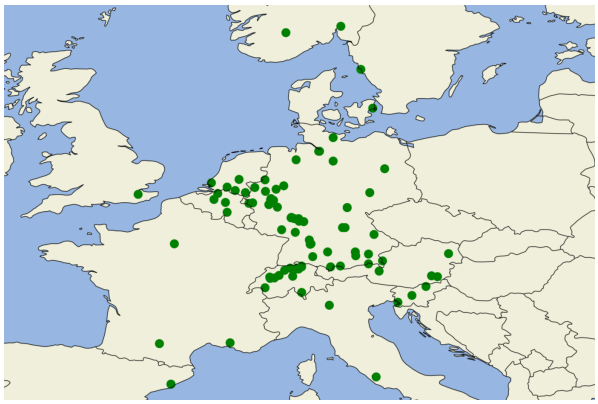

Figure 4. The geographical distribution of the Radarcape sensors used for the analysis.

5. Experimental Setup

5.1 Historical Analysis

Our historical analysis is based on real-world data collected by OpenSky in the 8-month period between July 2022 and February 2023. The analyzed 8-month data set contains 83 Terabytes of data.

5.2 Spoofing Attacks

For ethical and legal reasons, we did not perform our attack scenarios in the real world but opted to simulate the effects of different types of attack on our data.

For our simulated attacks, we consider a sensor to be in range of a transmission if its Euclidean distance to the sender is less than 250 km, disregarding line-of-sight considerations. We set the probability of an ADS-B message being registered by a sensor within range to 70%. To model inaccuracies in accordance with our system model, we add a normally distributed random variable to each simulated time-of-arrival value, with a zero-mean and a standard deviation of 100 ns.

5.2.1 ADS-B Spoofing

The simulated ADS-B spoofing attacks were conducted by taking the raw data for the month of February 2023, choosing 1% of all flight tracks at random, and modifying the time-of-arrival data of the associated records to be consistent with a stationary attacker.

To model the attacker, we fixed their location along a point of the claimed flight track, and simulate time-of-arrival data at nearby sensors using the parameters above. This approach models airborne attacks, for example using a drone stationed at an altitude with line-of-sight to multiple receivers.

5.2.2 GNSS Spoofing

To simulate GNSS spoofing, we model an attacker who leads an aircraft off its intended course by spoofing its GNSS receiver, similar to real-world incidents in the Middle East [12, 13]. Our simulations are based on data from February 2023. We took one hour batches, chose 1% of flight tracks, and modified them to simulate a straight flight track for the first 20% of the flight, then turning left by 20 degrees due to GNSS spoofing, and flying straight ahead on this new trajectory for the remaining time. The ADS-B location data is modified to hide any turn being made, keeping straight ahead on the original trajectory. To achieve reasonable separation from the original flight track, we have only considered flight tracks with more than 1000 received location messages.

5.3 Sensor Selection

As TDoA-based location verification requires highly precise clocks, we therefore chose the Radarcape sensor type with a timestamp precision in the range of 50 ns as our basis. Figure 4 shows the locations of the sensors with sufficient density in Central Europe.

While we focused on Radarcapes due to computational constraints, it is possible to include other sensor types (notably, Raspberry Pi/dump190-based setups) and use the implemented sensor filtering method to identify all sensors working with sufficiently high precision. [19, 20]

6. Historical OpenSky Data Analysis

We have identified several phenomena that we attribute to different root causes with high confidence.

As a baseline, consider Figure 5 (left), in which the expected TDoA values of two sensors closely match the measured values after adjusting for a constant offset. This is what we would expect for a flight track and a pair of sensors which all report accurate and precise information.

6.1 Sensor Issues

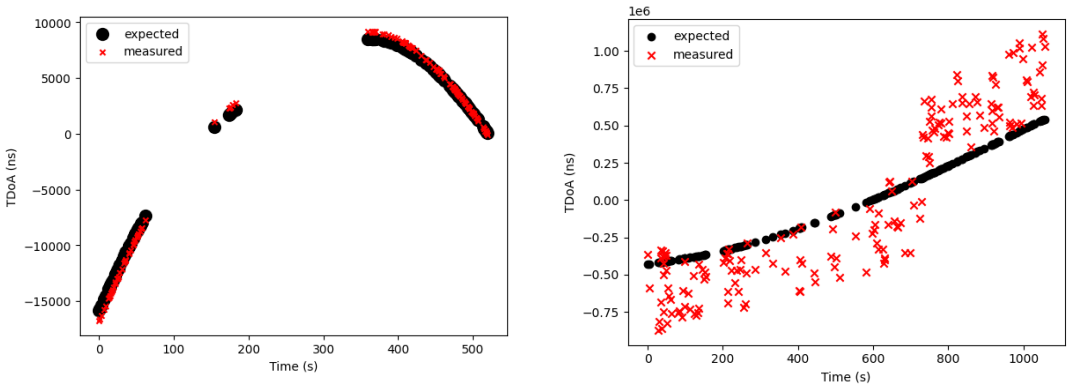

Figure 5. Example of good data (left). Sensor with bad clock precision (right).

We can detect sensor issues by comparing data from multiple sensor pairs for a given segment of a flight track. If only the TDoA graphs that involve a particular sensor show anomalous data, while the rest of the TDoA graphs look normal, we can infer that the sensor is problematic, if we have enough supporting data.

The most commonly observed root cause of sensor data errors was bad timestamp precision, which shows up in TDoA graphs as a wide band in the measured values. Often, this issue exhibits a checkerboard-like pattern, hinting at bad clock resolution, as seen in Figure 5 (right).

Another regularly observed issue seems to stem from sensors which reported their own location incorrectly. If TDoA graphs which include a particular sensor consistently show erroneous data within a narrow band, while other sensor-pairs report data consistent with the advertised flight track, we conclude that this sensor is misreporting its location, and filter it out for subsequent analysis steps. An example of such a finding is shown in Figure 6 (left). With sufficient data volume, it would be possible to approximate the sensor's true position using multilateration techniques, and continue to use it for further analysis using the corrected location. [16]

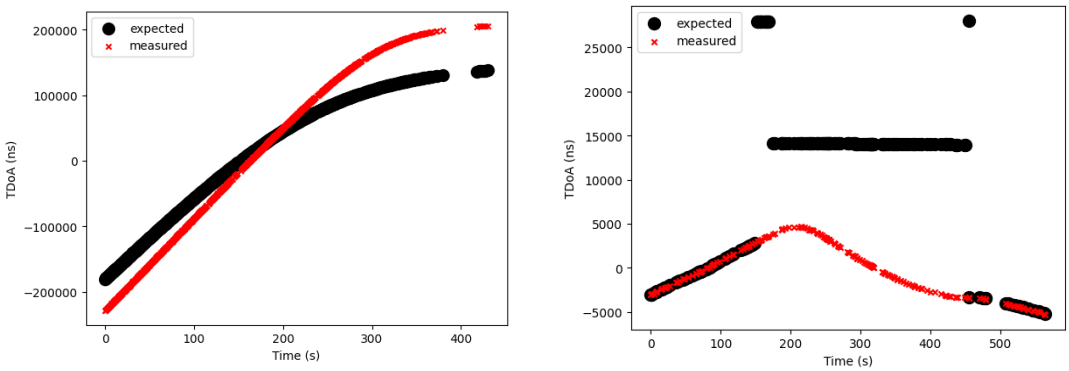

Figure 6. Sensor reporting wrong location (left). Temporary localization failure (right).

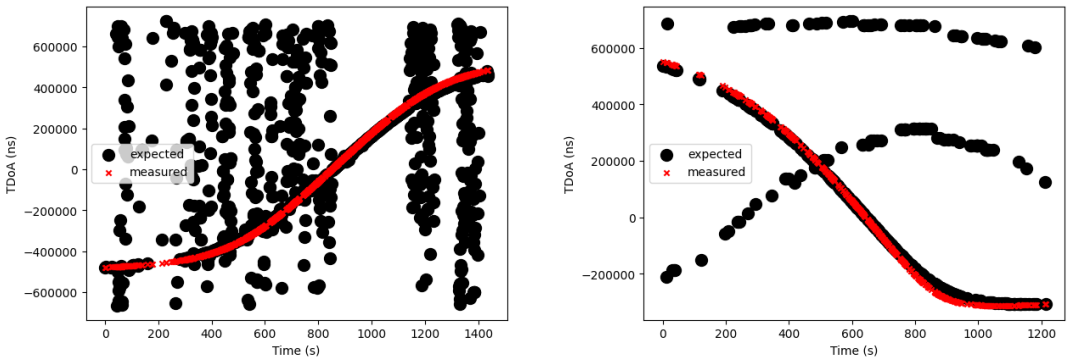

Figure 7. Malfunctioning aircraft equipment (left). Unidentified data error (right).

6.2 Real-World ADS-B Data Issues

In contrast to sensor problems, issues caused by erroneous ADS-B data create off-nominal graphs across most or all sensor pairs, which receive the incorrect data. In our analysis, we were able to identify multiple classes of issues, some of which were previously discussed in [21].

One aircraft exhibited a pattern of sending location data that was mostly inaccurate, often by a large margin (see Figure 7, left)). This behavior could indicate a GNSS jamming attack, where the aircraft receives false location data and only occasionally gets its true location, if it can lock on to the satellites despite the jamming. However, further analysis revealed that this aircraft showed this behavior consistently over long time spans and across wide geographical areas. This suggests that the aircraft has defective equipment rather than the aircraft being under attack.

In another class of problematic ADS-B downlink information, aircraft send wrong location data, but only for limited periods of time, and the advertised location barely changes from one message to the next. One such example is depicted in Figure 6 (right). We were unable to conclusively determine a root cause for this type of failure. Some possible causes include temporary loss of GNSS signal or a software glitch within the aircraft's systems, though we cannot rule out that the aircraft's GNSS receiver has been spoofed during the observed time interval.

Another issue we have observed is particularly interesting. Here, the expected values on the TDoA graphs look like multiple aircraft on different trajectories are overlaid, while the measured values are consistent with a single flight track, and match one of the seemingly overlaid flight tracks (see

Figure 7, right). Further investigation showed evidence that there were indeed two aircraft flying over central Europe using the same ICAO identifier. We suspect that the location messages from the other aircraft could have caused issues with the CPR location decoding algorithm [22]. One explanation is that the correct track is decoded if both, the even and odd CPR frames, stem from the tracked aircraft, and the two "ghost"-tracks are produced if either an even or an odd CPR frame from the other aircraft was received by another sensor, leading to the next complementary frame from the tracked aircraft decoding to a wrong location.

7. Detection of Simulated Spoofing Attacks

7.1 ADS-B Spoofing

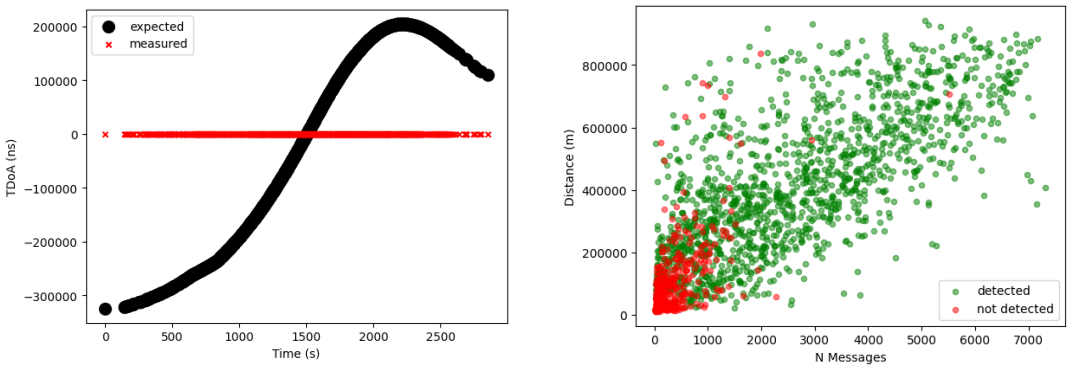

Figure 8. Simulated stationary ADS-B injection attack (left). Attack detection improves with more received messages, and greater distance covered (right).

In our ADS-B spoofing data sets, a total of 216 1-hour data batches over nine days in February 2023 were analyzed. The 1-hour data sets tracked a total of 198,802 flights, of which 1964 were randomly chosen and modified to simulate a stationary attacker, keeping the same claimed flight path as the original flight. An example of generated TDoA data is given in Figure 8 (left).

For the following evaluation, we only consider flight tracks, both original and modified, only if at least two legitimate sensors received a common subset of the flight track's messages. This leaves us with 1763 remaining spoofed tracks to be analyzed.

Overall, 81.28% of attacks were detected, with the detection rate increasing to 97.10% for flight tracks with over a thousand messages. Figure 8 (right) shows how the detection accuracy increases with the number of received messages, and total distance covered by a flight track.

Of the unmodified flight tracks, 154 flight tracks (0.08%) were falsely tagged as problematic. We consider these false positives, though in reality, some of these flight tracks exhibit clear deficiencies as outlined in the previous section.

7.2 GNSS Spoofing

GNSS spoofing attacks are more difficult to detect due to the more subtle change in TDoA measurements when compared to a stationary attacker who broadcasts wrong ADS-B data. Figure 9 (left) shows a sample TDoA graph of a GNSS spoofing attack.

The detection accuracy for our simulated GNSS spoofing attacks are significantly worse than for a stationary attacker injecting ADS-B signals into the network. 110 hours of data were analyzed for this scenario, with a total of 116,377 flight tracks, 1116 of which were modified to simulate GNSS

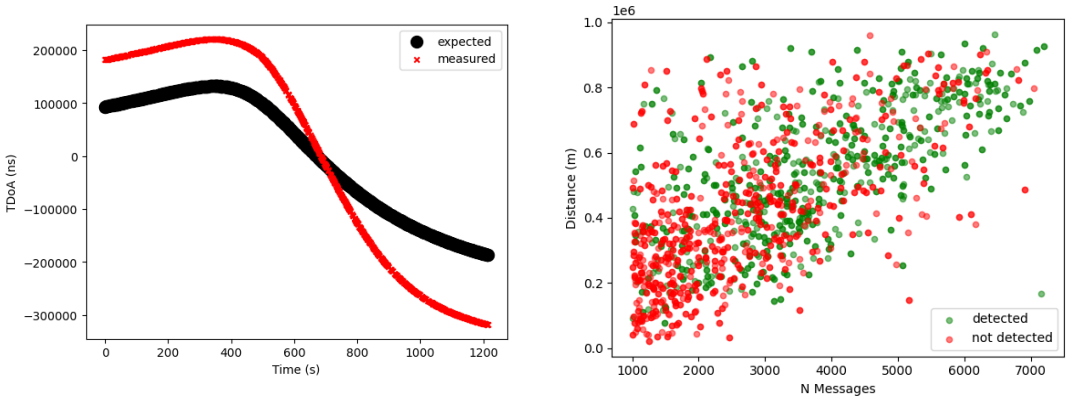

Figure 9. TDoA data of a simulated plane diverted off its advertised flight track at an angle of 20 degrees (left). Sophisticated GNSS spoofing attacks can be hard to differentiate from other types of data issues (right).

spoofing. 1072 of these could be analyzed, meaning some parts of these tracks were received by multiple receivers. Only 47.95% of the attacks were detected, with 14 unmodified flight tracks appearing over the detection threshold, a false positive rate of around 0.01%. As depicted in Figure 9 (right), detection accuracy increases with the number of messages and distance covered, but the increase is smaller compared to the simulated ADS-B spoofing attacks.

The lower detection rate of the simulated GNSS spoofing attacks when compared with stationary ADS-B spoofing attacks was expected, as the deviations between the expected and measured TDoA values are significantly smaller. Additionally, the deviation between expected and measured TDoA patterns of a plane flying at an angle to the advertised flight track closely resemble the patterns of a misplaced sensor, which can make it difficult to consistently differentiate between these two cases, especially when only few sensors receive relevant data.

8. Discussion

8.1 Computational Limitations

Over 2 billion records were collected each day within our geographical bounds, totalling 83 Terabytes of data over the 8-month period we analyzed.

Our final implementation needs roughly 5 minutes to analyze air traffic data generated in one hour. This shows that real-time implementation is possible, even if the region under analysis is significantly expanded and computationally more expensive methods used.

8.2 False Positives

Minimizing false positives presents a challenge, especially when using hardware with uncontrolled and unknown properties. With over 1000 flight tracks analyzed every hour, a 1% false positive rate would amass over 58'000 false alarms over the 8 months of our historical analysis.

We found numerous patterns in abnormal sensor behavior and have tuned our software to eliminate some of the false positives generated by faulty sensors. However, these decisions always have to be made carefully, as adversarial actions might sometimes create patterns similar to faulty sensors. Distinguishing these reliably in an automated fashion remains an open problem.

An example is inaccurate information about a sensor's location. This causes TDoA patterns very

similar to an aircraft which has been diverted off track by a GNSS spoofing attack (compare Figures 6 (left) and 9 (right)). If sufficient data is available, we can decide with some confidence whether this pattern emerges out of a spoofing attack or is just a result of wrong sensor data.

8.3 Possible Improvements

In order to include enough sensors to achieve meaningful large-scale geographical coverage, we had to set our detection thresholds for bad sensors and tracks so high that certain attack scenarios could not be detected. If, instead of filtering sensors based on a global threshold, we assess the TDoA measurement precision of each sensor pair, we could include this information in the final track verification step. This would allow a pair of high-quality sensors to detect more subtle attacks that would fall below our current global detection threshold. Similarly, a pair of low-quality sensors could be used to detect obvious attacks in regions not covered by higher-quality sensors. This change would increase the geographical coverage of our detection system and improve detection accuracy in regions with high-quality sensors, but would incur a significant increase in computational cost.

An additional way to improve the usefulness of our data would be to detect and correct incorrectly reported sensor information. Given enough TDoA measurements, multilateration techniques could be used to estimate the actual location of an affected sensor, allowing it to contribute useful information to our detection system. [16]

8.4 Applications

While our results show that sophisticated attackers can evade the detection mechanisms we have used for our analysis, they also demonstrate that most unsophisticated attacks can be detected without the need to set up and maintain dedicated hardware on an international scale. If our TDoA-based method is complemented with other approaches, such as using Kalman filters to sanity-check flight paths [23], or using a machine-learning based approach to location verification as described by Jansen et al. [6], detection accuracy could be further improved.

As new secure protocols are not planed to be implemented any time soon, and cheap, software-based devices are lowering the financial and skill hurdles to perform potentially severe attacks on ATC systems, the capability of crowdsourcing air traffic security could prove to be a valuable asset in the near future.

9. Conclusion

In this paper, we introduced a statistical location verification method for crowdsourced aircraft data. We implemented our method on the backend of a real-world sensor network, OpenSky, and showed that it is feasible to run in faster than real time. We evaluated it on 8 months of data collected from Radarcape sensors and demonstrated the effectiveness of our method in finding anomalous flight data, although we did (fortunately) not find clear evidence of widespread malicious spoofing attacks captured by OpenSky sensors. Simulated ADS-B and GNSS attacks were detected successfully with few false positives.

Author contributions

- L.B.: Visualization, Investigation, Methodology, Software, Writing–Original draft
- P.S.: Supervision, Methodology, Writing–Revised draft
- V.L.: Conceptualization, Development of OpenSky
- M.S.: Supervision, Conceptualization, Methodology, Writing–Revised draft

Open data statement

Supplementary data is available at Zenodo [24], illustrating example results and visualizations. Full data is available at the OpenSky Network.

References

[1] Xavier Olive, Axel Tanner, Martin Strohmeier, Matthias Schäfer, Metin Feridun, Allan Tart, Ivan Martinovic, and Vincent Lenders. "OpenSky Report 2020: Analysing in-flight emergencies using big data". In: *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*. IEEE. 2020, pp. 1–10.

[2] *After ADS-B launch, security concerns raised*. https://www.ainonline.com/aviation-news/aviation-international-news/2006-09-14/after-ads-b-launch-security-concerns-raised. Accessed on 2023-04-09.

[3] Andrei Costin and Aurélien Francillon. "Ghost in the Air (Traffic): On insecurity of ADS-B protocol and practical attacks on ADS-B devices". In: *black hat USA* 1 (2012), pp. 1–12.

[4] Martin Strohmeier, Giorgio Tresoldi, Leeloo Granger, and Vincent Lenders. "Building an avionics laboratory for cybersecurity testing". In: *Proceedings of the 15th Workshop on Cyber Security Experimentation and Test*. 2022, pp. 10–18.

[5] Martin Strohmeier, Ivan Martinovic, and Vincent Lenders. "Securing the air–ground link in aviation". In: *The Security of Critical Infrastructures: Risk, Resilience and Defense* (2020), pp. 131–154.

[6] Kai Jansen, Liang Niu, Nian Xue, Ivan Martinovic, and Christina Pöpper. "Trust the Crowd: Wireless Witnessing to Detect Attacks on ADS-B-Based Air-Traffic Surveillance". In: *NDSS*. 2021.

[7] Kai Jansen, Matthias Schäfer, Daniel Moser, Vincent Lenders, Christina Pöpper, and Jens Schmitt. "Crowd-gps-sec: Leveraging crowdsourcing to detect and localize gps spoofing attacks". In: *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2018, pp. 1018–1031.

[8] Martin Strohmeier, Vincent Lenders, and Ivan Martinovic. "Lightweight location verification in air traffic surveillance networks". In: *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security*. 2015, pp. 49–60.

[9] Martin Strohmeier, Ivan Martinovic, and Vincent Lenders. "A k-NN-based localization approach for crowdsourced air traffic communication networks". In: *IEEE Transactions on Aerospace and Electronic Systems* 54.3 (2018), pp. 1519–1529.

[10] *OpenSky Network*. https://opensky-network.org/. Accessed on 2023-04-09.

[11] Jahshan Bhatti and Todd E Humphreys. "Hostile control of ships via false GPS signals: Demonstration and detection". In: *NAVIGATION: Journal of the Institute of Navigation* 64.1 (2017), pp. 51–66.

[12] Joshua Kupietzky. *Why 20 Aircraft Went Off Course Over Iraqi Airspace*. https://simpleflying.com/20-aircraft-went-off-course-iranian-airspace/. Accessed on 2023-10-09.

[13] Matt Berg. *Israel's using widespread GPS tampering to deter Hezbollah's missiles*. https://www.politico.com/news/2023/10/23/israels-gps-tampering-deter-hezbollahs-missiles-00123026. Accessed on 2023-10-24.

[14] Michael Felux, Benoit Figuet, Manuel Waltert, Patric Fol, Martin Strohmeier, and Xavier Olive. "Analysis of GNSS disruptions in European Airspace". In: *Proceedings of the 2023 International Technical Meeting of The Institute of Navigation*. 2023, pp. 315–326.

[15] Martin Strohmeier, Xavier Olive, Jannis Lübbe, Matthias Schäfer, and Vincent Lenders. "Crowdsourced air traffic data from the OpenSky Network 2019–2020". In: *Earth System Science Data* 13.2 (2021), pp. 357–366.

[16] Yago Lizarribar, Domenico Giustiniano, Gérôme Bovet, and Vincent Lenders. “SkyPos: Real-world evaluation of self-positioning with aircraft signals for IoT devices”. In: *IEEE Journal on Selected Areas in Communications* (2023).

[17] Matthias Schäfer, Martin Strohmeier, Vincent Lenders, Ivan Martinovic, and Matthias Wilhelm. “Bringing up OpenSky: A large-scale ADS-B sensor network for research”. In: *IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*. IEEE. 2014, pp. 83–94.

[18] Daniel Moser, Patrick Leu, Vincent Lenders, Aanjhan Ranganathan, Fabio Ricciato, and Srdjan Capkun. “Investigation of multi-device location spoofing attacks on air traffic control and possible countermeasures”. In: *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. 2016, pp. 375–386.

[19] Martin Strohmeier, Mauro Leonardi, Sergei Markochev, Fabio Ricciato, Matthias Schäfer, and Vincent Lenders. “In Pursuit of Aviation Cybersecurity: Experiences and Lessons From a Competitive Approach”. In: *IEEE Security & Privacy* (2023).

[20] Martin Strohmeier, Mauro Leonardi, Sergei Markochev, Fabio Ricciato, Matthias Schäfer, and Vincent Lenders. “Improving Aircraft Localization: Experiences and Lessons Learned from an Open Competition”. In: *arXiv preprint arXiv:2209.13669* (2022).

[21] Matthias Schäfer, Martin Strohmeicr, Matthew Smith, Markus Fuchs, Vincent Lenders, and Ivan Martinovic. “OpenSky report 2018: Assessing the integrity of crowdsourced mode S and ADS-B data”. In: *2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC)*. IEEE. 2018, pp. 1–9.

[22] Junzi Sun. *The 1090 Megahertz Riddle: A Guide to Decoding Mode S and ADS-B Signals*. 2nd ed. TU Delft OPEN Publishing, 2021. ISBN: 978-94-6366-402-8. DOI: 10.34641/mg.11.

[23] Jimmy Krozel, Dominick Andrisani, Mohammad Ayoubi, Takayuki Hoshizaki, and Chris Schwalm. “Aircraft ADS-B data integrity check”. In: *AIAA 4th aviation technology, integration and operations (ATIO) Forum*. 2004, p. 6263.

[24] Martin Strohmeier, Patrick Schaller, and Lukas Baege. *Supplementary Material to "Scaling the Timing- Based Detection of Anomalies in Real-World Aircraft Trajectories"*. Oct. 2023.