5-2015

Challenges of Implementing Automatic Dependent Surveillance Broadcast in the Nextgen Air Traffic Management System

Carl J. Giannatto Jr.

CHALLENGES OF IMPLEMENTING AUTOMATIC DEPENDENT SURVEILLANCE BROADCAST IN THE NEXTGEN AIR TRAFFIC MANAGEMENT SYSTEM

By

Carl J. Giannatto, Jr.

B.A. University of South Florida, 1988

A THESIS

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

(in Computer Science)

The Graduate School

The University of Maine

May 2015

Advisory Committee:

George Markowsky, Professor of Computer Science, Advisor

Yifeng Zhu, Associate Professor of Electrical and Computer Engineering

Linda Markowsky, Assistant Research Professor of Computer Science

THESIS ACCEPTANCE STATEMENT

On behalf of the Graduate Committee for Carl J. Giannatto, Jr., I affirm that this manuscript is the final and accepted thesis. Signatures of all committee members are on file with the Graduate School at the University of Maine, 42 Stodder Hall, Orono, Maine.

Dr. George Markowsky, Professor of Computer Science Date

LIBRARY RIGHTS STATEMENT

In presenting this thesis in partial fulfillment of the requirements for an advanced degree at The University of Maine, I agree that the Library shall make it freely available for inspection. I further agree that permission for "fair use" copying of this thesis for scholarly purposes may be granted by the Librarian. It is understood that any copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Signature:

Date:

CHALLENGES OF IMPLEMENTING AUTOMATIC DEPENDENT SURVEILLANCE BROADCAST IN THE NEXTGEN AIR TRAFFIC MANAGEMENT SYSTEM

By Carl J. Giannatto, Jr.

Thesis Advisor: Dr. George Markowsky

An Abstract of the Thesis Presented
in Partial Fulfillment of the Requirements for the
Degree of Master of Science
(in Computer Science)
May 2015

The Federal Aviation Administration is in the process of replacing the current Air Traffic Management (ATM) system with a new system known as NextGen. Automatic Dependent Surveillance-Broadcast (ADS-B) is the aircraft surveillance protocol currently being introduced as a part of the NextGen system deployment. The evolution of ADS-B spans more than two decades, with development focused primarily on increasing the capacity of the Air Traffic Control (ATC) system and reducing operational costs. Security of the ADS-B communications network has not been a high priority, and the inherent lack of security measures in the ADS-B protocol has come under increasing scrutiny as the NextGen ADS-B implementation deadline draws near.

The research conducted in this thesis summarizes the ADS-B security vulnerabilities that have been under recent study. Thereafter, we survey both the theoretical and practical efforts which have been conducted concerning these issues, and review possible security solutions. We create a classification of the ADS-B security solutions considered and provide a ranking of the potential solutions. Finally, we discuss the most compatible approaches available, given the constraints of the current ADS-B communications system and protocol.

DEDICATION

To my wife and best friend Sharon, thank you for always being there. I would never have completed this endeavor without your incredible patience and support.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my advisor, Dr. George Markowsky, whose collaboration, guidance and support made this thesis possible. I would also like to thank Dr. Linda Markowsky and Dr. Yifeng Zhu for their assistance and suggestions in completing this work. Additionally, I would like to thank Maj Jesse Salisbury, ANG for introducing me to the topic of this research. Furthermore, I would like to acknowledge Maj Gen Maurice Kendall, USA (Ret.), LtCol Bud Barbee, USAF (Ret.) and Capt. Wayne Cheal, USN (Ret.), without whose recommendations, advice and encouragement I would never have started my aviation career so many years ago. Finally, I would like to thank my parents for instilling in me the work ethic, values and moral compass that guide me to this day.

TABLE OF CONTENTS

DEDICATION iii

ACKNOWLEDGEMENTS iv

LIST OF TABLES viii

LIST OF FIGURES ix

LIST OF ABBREVIATIONS x

CHAPTER 1: INTRODUCTION 1

General Issue 1

Problem Statement 2

Research Objectives 3

CHAPTER 2: AIR TRAFFIC MANAGEMENT SYSTEM OVERVIEW 4

Air Traffic Control System History 4

Air Traffic Control System Today 5

NextGen and Automatic Dependent Surveillance-Broadcast 7

ADS-B System Description 9

CHAPTER 3: ADS-B VULNERABILITIES 14

Eavesdropping 15

Jamming 15

Message Injection 16
Message Deletion 17
Message Modification 18
CHAPTER 4: ADS-B SECURITY REQUIREMENTS 19
ADS-B Network Properties 19
Required Security Attributes 21
ADS-B Security Solutions Taxonomy 23
CHAPTER 5: SECURE BROADCAST AUTHENTICATION SOLUTIONS 26
Cryptographic Schemes 27
Public Key Infrastructure 30
Retroactive Key Publication 32
Aircraft Address Message Authentication Code 36
Non-Cryptographic Schemes 40
Physical Layer 40
Spread Spectrum 42
CHAPTER 6: SECURE LOCATION VERIFICATION SOLUTIONS 45
In-Region Verification 45
Distance Bounding 46
Kalman Filtering for Intent Verification 48

Secure Location Determination 51

Multilateration 52

Data Fusion 56

CHAPTER 7: ANALYSIS OF SECURITY SOLUTIONS 60

Cost-Effective Solutions 60

Evaluation of Scheme Implementation Considerations 61

Evaluation of Scheme Security Provided 65

Evaluation of Scheme Message Integrity Provided 67

Summary of ADS-B Security Schemes 69

CHAPTER 8: FUTURE WORK & CONCLUSION 72

NextGen Future Research 73

Post-NextGen Future Research 74

Conclusion 77

REFERENCES 78

APPENDIX A: AA-MAC TEST PROGRAM 82

APPENDIX B: SINGLE-VARIABLE KALMAN FILTER PROGRAM 85

BIOGRAPHY OF THE AUTHOR 88

LIST OF TABLES

Table 1. Comparison of Transponder Modes to ADS-B [7] 10

Table 2. Scheme Implementation Considerations 64

Table 3. Scheme Security Provided 66

Table 4. Scheme Features Provided 68

Table 5. Scheme Ranking Summary 69

LIST OF FIGURES

Figure 1. ADS-B protocol hierarchy [6] 10

Figure 2. Overview of ADS-B system architecture [7] 11

Figure 3. 1090ES Data Link Message Format 12

Figure 4. Taxonomy of ADS-B Security [7] 24

Figure 5. Example of µTESLA time-released key chain for source authentication [17] 35

Figure 6. ADS-B message 24-bit PI field replaced by 24-bit MAC 37

Figure 7. Principle of distance bounding protocols [7] 46

Figure 8. Single-variable Kalman filtering example 49

Figure 9. Basic MLAT architecture [7] 52

Figure 10. Intersection of Three TDOA Hyperboloids [21] 53

Figure 11. Construction of TSOA Ellipse [21] 54

Figure 12. ADS-B/SSR Fusion Model [23] 57

LIST OF ABBREVIATIONS

AADS:	Airplane Asset Distribution System
AA-MAC:	Aircraft Address Message Authentication Code
ADS-B:	Automatic Dependent Surveillance – Broadcast
ADS-R:	Automatic Dependent Surveillance – Rebroadcast
ARTCC:	Air Route Traffic Control Center
ATC:	Air Traffic Control
ATM:	Air Traffic Management
CA:	Certificate Authority
CIR:	Channel Impulse Response
DoD:	Department of Defense
DoS:	Denial of Service
DSSS:	Direct Sequence Spread Spectrum
ES:	Extended Squitter
FAA:	Federal Aviation Administration
FHSS:	Frequency Hopping Spread Spectrum
FSS:	Flight Service Station
GNSS:	Global Navigation Satellite System
GPS:	Global Position System
ICAO:	International Civil Aviation Organization
LOS:	Line of Sight

MAC:	Message Authentication Code
MANET	Mobile Area Network
MHz:	Megahertz
MLAT	Multilateration
NAS:	National Airspace System
NAVAID:	Navigational Aid
NM:	Nautical Mile
PSR:	Primary Surveillance Radar
RF:	Radio Frequency
RKP:	Retroactive Key Publication
RSS:	Received Signal Strength
SSR:	Secondary Surveillance Radar
STARS:	Standard Terminal Automation Replacement System
TDOA:	Time Difference of Arrival
TESLA:	Timed Efficient Stream Loss-Tolerant Authentication
TRACON:	Terminal Radar Approach Control
TSOA:	Time Sum of Arrival
UAT:	Universal Access Transceiver
UAV:	Unmanned Aerial Vehicle
UG3d:	Upgraded Third Generation
VHF:	Very High Frequency
WAM	Wide Area Multilateration

CHAPTER 1

INTRODUCTION

Over the past two decades, the Federal Aviation Administration (FAA) has been working on a replacement for the current air traffic control (ATC) system in a project known as NextGen. Developed in cooperation with other aviation agencies, the goal of NextGen is to shift air traffic surveillance and management technology away from an infrastructure based on radar to one that obtains position information from a Global Navigation Satellite System (GNSS). This surveillance paradigm shift offers the potential to reduce deployment and maintenance costs, while at the same time increase both the capacity and safety of the global air traffic system.

General Issue

The new ATC surveillance system being deployed as part of NextGen is called Automatic Dependent Surveillance-Broadcast (ADS-B). The key issue with ADS-B is that it was not developed with security as a priority, leaving it susceptible to a number of different radio frequency (RF) attacks. Recent research has demonstrated the ease of compromising the security of ADS-B using inexpensive Universal Software Radio Peripheral (USRP) hardware and Open Source software [1], [2].

These vulnerabilities are generating increasing concern as the deadline for full compliance by the aviation industry draws near. The European Aviation Safety Agency (EASA) has mandated all aircraft in European airspace be equipped with ADS-B by 2017,

while the FAA has set 2020 as its implementation target. In addition, countries such as Australia have already deployed full continental coverage, with ADS-B surveillance being the sole means of ATC in sparsely populated regions of the country. Although aviation agencies previously estimated that 70-80 percent of commercial aircraft worldwide would be equipped with ADS-B by 2013 [3], a recent report by the Department of Transportation's Inspector General [4] indicates that compliance within the aviation industry is running behind schedule. The report cites concerns over system vulnerabilities as one of the principle causes for fleet-wide delays in ADS-B equipment installation.

Problem Statement

The implementation of a new aircraft surveillance system is a non-trivial, decades-long process that has far reaching implications on all segments of the aviation industry. Not only is there a substantial cost in developing and deploying the system infrastructure, there are significant synergies required within the industry to train flight crews and air traffic controllers on the use of the new system.

As reported in [4], the security shortcomings in ADS-B are creating uncertainty within the aviation community and reluctance toward making a commitment to complying with the NextGen deployment plan. As the timetable for the scheduled implementation of ADS-B grows shorter, solutions to address the vulnerabilities in ADS-B must be found. Potential approaches must be evaluated from both a security and a cost standpoint. Therefore, feasible solutions must strike a balance between security

improvement and compatibility with the current ADS-B communications system.

Research Objectives

The purpose of this research is to review the strengths and weaknesses of proposed ADS-B security schemes, considering both the security offered by the scheme and its compatibility with the current air traffic management (ATM) system infrastructure. We begin with a discussion of the ATM system in Chapter 2, giving an overview of the evolution of our ATC system, NextGen and the ADS-B protocol. In Chapter 3 we present a summary of the ADS-B vulnerabilities that have been discussed in the recent literature. Building on this discussion, a model of the ADS-B network is outlined in Chapter 4 and the required security attributes of the network are identified. In Chapters 5 and 6, we discuss the various ADS-B security proposals. We conclude our research in Chapter 7, where we develop a ranking system to evaluate the various security schemes and identify the most beneficial approaches, considering their security features and cost-effectiveness of the various proposals.

CHAPTER 2

AIR TRAFFIC MANAGEMENT SYSTEM OVERVIEW

Air Traffic Control System History

Our current air navigation and ATC systems trace their origins back to the 1920s. During this early period in air navigation, the Post Office Department utilized lighted beacons as a navigational aid to pilots flying postal delivery aircraft at night. In the 1930s these visual aids were replaced by non-directional radio beacons (NDBs), which transmit pulses of electromagnetic energy modulated with Morse Code.

By the late 1930s commercial air travel was becoming a popular mode of transportation and the volume of air traffic increased dramatically. As it became more difficult to keep track of the increasing number of aircraft in operation, the airlines developed a system of radio stations to help monitor their en route air traffic. These initial radio stations were located in Chicago, Newark and Cleveland and were the precursor to our current air traffic control system. The Bureau of Air Commerce acquired the radio stations in 1936 and in so doing formed what is considered the First Generation of ATC.

This First Generation ATC system consisted of no automation and very little radar coverage. The fledgling ATC system relied on manual methods of tracking aircraft using progress strips for each flight. By the late 1950s the volume of aircraft in operation had

increased to the point that manual tracking was no longer feasible. In 1959 the Second Generation ATC system was introduced, which automated many of the flight monitoring tasks through the use of computers for processing air traffic data and ground based radar to help track individual aircraft. Two years later, another major improvement to the ATC system was made when the FAA incorporated ground based equipment to interrogate a transponder located on the aircraft, allowing each air traffic radar target to be uniquely identified.

In the late 1960s, air traffic was again taxing the capabilities of the National Airspace System (NAS). By the early 1970s, advances in computer technology made it possible for Upgraded Third Generation development (UG3d) of the ATC system. UG3d provided the FAA with the ability to upgrade equipment used in both the terminal and en route air traffic control structures. Through the increased automation of controller tasks and the ability to receive timely flight tracking information, UG3d enabled air traffic controllers to safely accommodate and monitor the increasing volume of air traffic.

Air Traffic Control System Today

With the exception of introducing Global Positioning System (GPS) technologies in the late 1990s, the current NAS infrastructure has undergone few changes since the improvements incorporated into UG3d. Currently the NAS consists of a large number of facilities including approximately 750 ATC installations, over 18,000 airports and more than 4,500 air navigation stations. The 750 ATC facilities are comprised of 21 Air Route

Traffic Control Centers (ARTCCs), 197 Terminal Radar Approach Control (TRACON) facilities, more than 450 airport control towers and numerous Flight Service Station (FSS) facilities.

ARTCCs are responsible for controlling en route traffic within designated control sectors, with the majority of the en-route traffic traveling along designated airways at and above 18,000 feet. TRACON facilities control aircraft within an approximate 30 nautical mile radius of the larger airports within the ATC system, while airport control towers are responsible for controlling aircraft within a 5 nautical mile radius of the airport. FSS facilities are auxiliary components of the ATC system and provide general information to pilots such as weather and traffic advisories.

Current NAS aircraft surveillance techniques fall into three basic categories: Procedural ATC, Primary Surveillance Radar (PSR) and Secondary Surveillance Radar (SSR). Procedural ATC is what is known as a *dependent surveillance* technique, which means it depends on input from individual aircraft. With Procedural ATC, pilots are required to periodically report their position using radio communications, and it is predominately used for oceanic and remote area flight operations where there is little or no radar coverage. PSR is a non-cooperative and *independent surveillance* system typically used by TRACON facilities and in busy terminal areas. These high definition radar systems determine aircraft position via target range and azimuth from the station and do not depend on any input from the aircraft. SSR is a cooperative and *partially-independent surveillance* system typically used for en-route tracking by ARTCCs. The

SSR radar system is a lower-resolution system than PSR, and determines aircraft position through a combination of radar target return and aircraft transponder reply when interrogated by a ground station.

Many of the current ATC facilities have been in service for more than 50 years. These installations, and in particular the ground-based SSR and PSR radar systems, are very costly to operate and maintain. Increased air traffic, aging equipment and a desire to leverage technological advancements necessitate a comprehensive overhaul to the NAS. In its current form, the air transportation system performs adequately but it is once again approaching its capacity limits. Without a makeover, the expected growth in air traffic will likely create costly flight delays and increased flight safety hazards.

NextGen and Automatic Dependent Surveillance-Broadcast

In response to these concerns, the FAA began the development of NextGen, which incorporates new technologies to meet anticipated future NAS demands. The primary goal of NextGen is to significantly increase the safety and capacity of the air traffic management system. The upgrade incorporates a fundamental conversion of the entire NAS, including the addition of satellite-based technologies for surveillance operations and the shutdown of many legacy ground-based radar systems currently in use. A key component of NextGen is the position reporting and tracking offered by ADS-B.

The ADS-B surveillance system is *automatic* in that it requires no pilot or controller intervention. It is *dependent surveillance* because the aircraft provides input to the air traffic control system based on information derived from the aircraft's GPS receiver. As a *broadcast* protocol, ADS-B will continually transmit an updated position and other data to nearby ground stations and aircraft on a regular interval. This broadcast occur every several hundred milliseconds, compared with PSR which updates aircraft position information once every 4 to 5 seconds. As a result, ADS-B provides much a higher surveillance rate and accuracy than PSR and SSR. For example, at distance of 60 nautical miles from the ground station, ADS-B provides ±20 meters of precision compared to ±300 meters offered by the SSR radar system.

ADS-B has the potential to improve safety through enhanced pilot and controller situational awareness, better inflight collision and runway incursion avoidance, and the ability to implement accurate ATC surveillance in remote geographic areas with no current radar coverage. Better position monitoring accuracy should allow the air traffic control system to handle a higher volume of aircraft through condensed aircraft separation standards, more direct traffic routings and optimized departures and approach procedures. Another potential benefit of the NextGen ADS-B infrastructure is a reduction in air traffic control system maintenance and operating costs, since the new system is comprised of simple UHF radio stations that are significantly cheaper to install and maintain than the aging surveillance radar ground stations [5].

ADS-B System Description

The FAA's NextGen implementation plan includes a network of approximately 800 ADS-B ground stations, placed 150 to 200 miles apart. These stations will receive signals from two competing ADS-B data link standards: Universal Access Transceiver (UAT) and Extended Squitter (1090ES). The UAT data link was specifically designed for ADS-B and has a much larger (272-bit) message data block than 1090ES (56-bit) in order to accommodate supplementary aviation services information. It establishes a channel with a data rate of 1 Mbps and operates at 978 MHz. The message format of UAT is incompatible with any existing ATM system protocol, and thus requires aircraft to be equipped with new avionics.

To minimize the cost impact on commercial and military aviation fleets, the FAA decided to employ a separate data link protocol based on an existing interrogation equipment mechanism in the SSR Mode S transponder called extended squitter. The term *squitter* refers to the periodic broadcast of aircraft tracking data. When a Mode S transponder is interrogated by SSR, its response to the interrogation message is called a *squawk*. The transponder also periodically sends out aircraft tracking data without being interrogated in what is called a *squit* transmission. The 1090ES protocol extends the original 56-bit Mode S message to 112-bits, hence the term *extended squitter.*

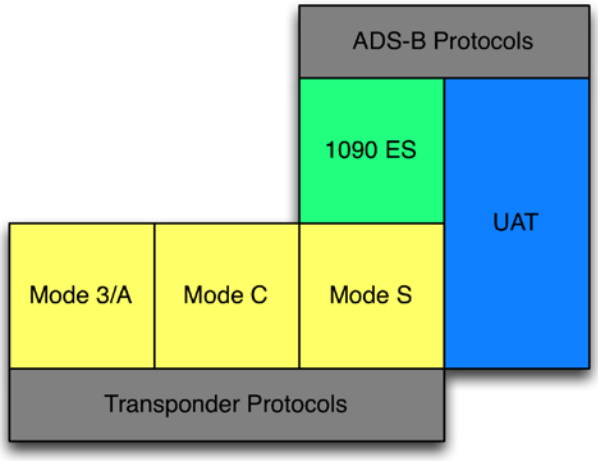

Figure 1. ADS-B protocol hierarchy [6].

The relationship between the Transponder and ADS-B protocols is shown in Figure 1. The purpose here is to show the relationship between the legacy transponder components Mode 3/A, Mode C and Mode S, as well as to emphasize that the 1090ES protocol is built on the existing Mode S protocol. It also demonstrates that the UAT protocol is a completely separate protocol from 1090ES. Table 1 shows the relative message sizes for the existing transponder protocols and the two ADS-B protocols.

	Message Length	Frequencies	Operational Mode	Use Cases
Mode A	12-bit	1030 / 1090 MHz	Independent / Non-selective interrogation	Target Identification
Mode C	12-bit	1030 / 1090 MHz	Independent / Non-selective interrogation	Target Pressure Altitude
Mode S	56 / 112-bit	1030 / 1090 MHz	Independent / Selective interrogation	Multiple
ADS-B / 1090ES	112-bit	1090 MHz	Dependent / Automatic	Multiple
ADS-B / UAT	420-bit	968 MHz	Dependent / Automatic	Multiple

Table 1. Comparison of Transponder Modes to ADS-B [7].

The 1090ES protocol is based on the traditional Mode S system and adds the message fields for ADS-B surveillance data, which allows the ADS-B function to be incorporated in current Mode S transponders. Since it is based on existing avionics equipment, the cost of equipping a fleet of aircraft with 1090ES is substantially less than it would be for purchasing entirely new UAT-compatible avionics.

ADS-B is separated into two functional operations; ADS-B OUT and ADS-B IN. ADS-B OUT is the continuous broadcast of aircraft position data along with identity, altitude, speed and rate of climb/descent. ADS-B IN is an optional service that allows properly equipped aircraft to receive and display detailed information on other aircraft operating in the same area (see Figure 2).

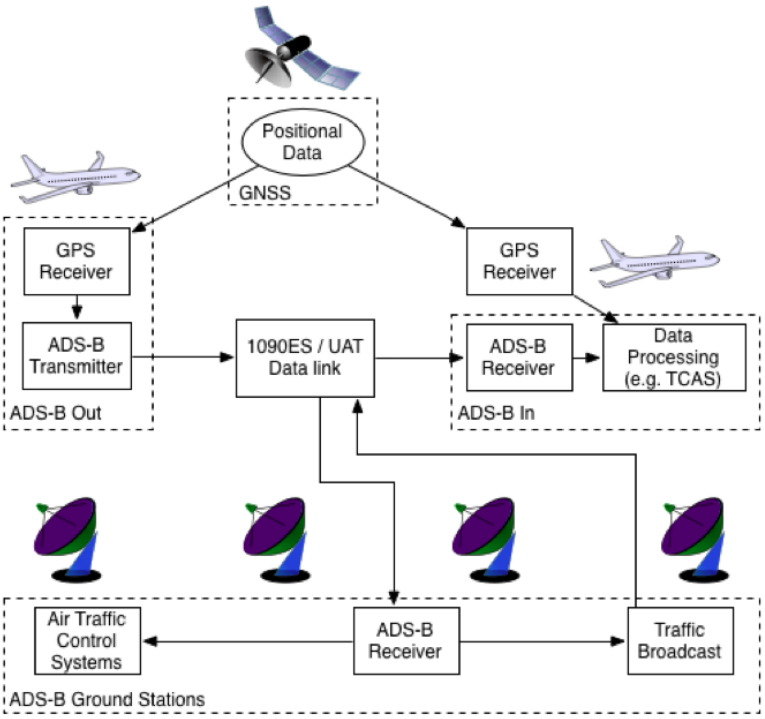

Figure 2. Overview of ADS-B system architecture [7].

To facilitate interoperability between aircraft using different frequencies, the system incorporates a support component called Automatic Dependent Surveillance-Rebroadcast (ADS-R). ADS-R receives the traffic information broadcasts on the 1090MHz or 978 MHz links and rebroadcasts the information to aircraft on the opposite data link frequency [8], [7]. Since the UAT protocol will primarily be used by general aviation aircraft, we will limit our discussion of ADS-B security solutions to the 1090ES protocol.

The 1090ES data link utilizes a standardized message format and transmission protocol, consisting of a preamble (consisting of two synchronization pulses) followed by a 112 bit message, as shown in Figure 3.

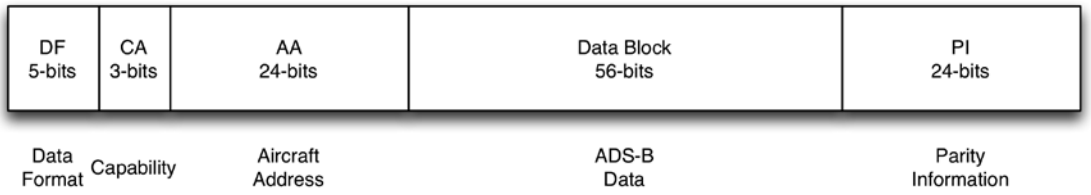

Figure 3. 1090ES Data Link Message Format.

The downlink format field DF (alternatively UF for uplink messages) assigns the type of the message. A downlink format value of 17 indicates that the message is an extended squitter, enabling the transmission of 56 arbitrary bits in the Data Block field. The CA field indicates information about the capabilities of the Mode S transponder, while the 24 bit AA field carries the unique International Civil Aviation Organization (ICAO) aircraft address which enables aircraft identification. Finally, the PI-field provides a 24 bit cyclic

redundancy check (CRC) to detect and correct possible transmission errors. Using the 24-bit parity information and a fixed generator polynomial of degree 24, it is possible for recipients to correct up to 5 bit errors in 1090ES messages [7]. This error correction limit is important, as any message exceeding 5 bit errors is dropped as a corrupt message. Currently the message drop rate in the ADS-B network is about 33%, so there are a significant number of bit errors occurring in the ADS-B messages. The majority of these errors appear to be the result of congestion on the ADS-B communication frequencies.

CHAPTER 3

ADS-B VULNERABILITIES

The initial development to extend the Mode S protocol for use in the ADS-B surveillance system was begun over 20 years ago [9]. At the time, the primary concern of the developers was on increasing the ATC surveillance system operational capacity, reliability, accuracy and range [10]. There was no emphasis on providing security to the new system, and as a result ADS-B contains many security weaknesses that potential attackers can exploit. These vulnerabilities inherently stem from the nature of broadcast communication when used without additional security measures. Unlike traditional point-to-point wired networks which present physical access barriers, there are no impediments for an attacker trying to access a wireless broadcast network. The security issues caused by the open nature of the ADS-B network are compounded by the fact that the messages are broadcast as unencrypted plaintext.

As a result of the broadcast characteristics and unencrypted message format of the network, access control mechanisms for ADS-B are very challenging to implement. Adding to the security problems caused by accessibility, recent work by Magazu [2] and Costin et al. [1] demonstrates that the widespread obtainability of inexpensive RF implementation hardware and software has facilitated the ability of hackers to design successful exploits. In the remainder of the chapter we present an overview of the various ADS-B passive and active attack vulnerabilities, discussed in increasing order of

difficulty and complexity [7].

Eavesdropping

Passive listening to the unsecured transmissions is the simplest and most direct form among the many security vulnerabilities present in ADS-B. Since ADS-B messages are sent plaintext over a broadcast communications network, the protocol's susceptibility to eavesdropping is well known and has been a topic of discussion since its early development. Although many aviation services and hobbyists gather and disseminate this information with non-nefarious intentions, reconnaissance through passive listening often forms the basis for a number of more sophisticated network attacks. By combining ADS-B provided data with other publicly available data sources (e.g. official databases provided by aviation authorities), attackers can retrieve enough information to launch targeted attacks [6]. On a broadcast network, eavesdropping is practically impossible to detect and is difficult to prevent without fully encrypting the data.

Jamming

Jamming is an active attack that is slightly more complex than eavesdropping, affecting either a single node or multiple nodes in an area of a wireless network. In a jamming attack an adversary disrupts the transmission and reception of messages by sending a sufficiently high-powered signal on the wireless frequency. While jamming is a problem common to all wireless communications, the impact on aviation is exacerbated by unrestricted access to system's wide open spaces as well as the time-

critical nature of the transmitted data.

The two basic categories of jamming attacks on ADS-B are Ground Station Flood Denial and Aircraft Flood Denial. The intent of both of these attacks is to disrupt the communications frequency and effectively block the surveillance network. Since an adversary can gain close proximity to a ground station, a Ground Station Flood Denial attack is the easier of the two for an adversary to employ. Jamming a ground station can be accomplished using much lower power on the frequency than is required to target an airborne node. Aircraft Flood Denial is slightly more difficult, as the adversary does not have ease of proximity to the target. A successful attack requires a much higher powered signal jamming device, and is most likely to pose a threat to landing and departing aircraft rather than en-route traffic.

Message Injection

Although slightly more difficult to conduct than jamming attacks, recent research by Magazu [2] and Costin et al. [1] detailed the relative ease of injecting non-legitimate messages into the air-traffic communication system using simple and readily available technology. Since no authentication measures are implemented at the data link layer, there is essentially no obstacle for an attacker in building a transmitter that is able to produce correctly modulated and formatted ADS-B messages [7]. As with jamming attacks, message injection attacks can target both ground-based and airborne targets, producing illegitimate ghost targets that appear as valid nodes to the network participants.

Message Deletion

Higher up on the difficulty scale are Message Deletion attacks, where legitimate messages are removed from the wireless network using either destructive or constructive interference. In a constructive interference attack, the adversary attempts to obscure the sender's transmission by causing a large number of bit errors. The theory behind a constructive interference attack is to cause a sufficient number of errors so that the receiver sees the message as corrupt and drops the message. Since Mode S extended squitters' CRC can correct a maximum of 5 bit errors per message, an adversary will be successful if they can cause a message to exceed this threshold.

In contrast to generating bit errors, a destructive interference attack tries to mask network communications messages by transmitting the inverse of the signal broadcast by a legitimate sender. The theory behind destructive interference is that by transmitting an inverse signal, the sender's signal will be highly attenuated and obscured. In practice, destructive interference is extremely challenging to implement, due to very precise and complex timing requirements. Unlike destructive interference, constructive interference does not require precise time synchronization and tends to be more effective. The end result of both of these attacks is that, from the perspective of the network participants, a node that was previously part of the network suddenly disappears [7].

Message Modification

The most difficult vulnerabilities to exploit are those involving ADS-B message modification. These attacks are complex to successfully implement because they typically require the attacker to access the ADS-B network communications hardware during message transmission, which is much more difficult to accomplish. There are two different approaches that an attacker can use for message modification: Overshadowing and Bit-Flipping. An attacker employs overshadowing by sending a high-powered signal that is precisely timed with the transmission of the target message. This has the effect of replacing the target message in whole or in part, allowing the sender's message to be modified or replaced entirely. When an adversary uses bit-flipping, the attacker converts any number of bits from 1 to 0 (or the other way around) by superimposing a false signal over the original signal. In both cases arbitrary data can be injected into the network without the knowledge of any of the participants. This effect can also be achieved by combining message deletion and injection, but message modification at the ADS-B network communications hardware level can be regarded as more problematic than the injection of a completely new message, since the manipulated message was originally considered legitimate by the network [7].

CHAPTER 4

ADS-B SECURITY REQUIREMENTS

The challenges to addressing security problems in ADS-B stem from its open broadcast architecture and the need for security schemes to integrate into the operational characteristics of the existing air traffic management system. In order to describe the security requirements of the ADS-B communications network, we begin by identifying the properties of the network as outlined by Strohmeier et al. [7] and then discuss the security attributes needed to adequately address the vulnerabilities in the system.

ADS-B Network Properties

The ADS-B network is a mobile ad hoc network (MANET), consisting of a large and variable number of highly mobile nodes moving at velocities of 500 mph[1] or more. Due to the speed and mobility of its nodes, the ADS-B network is extremely dynamic, with very short duration communications between nodes. Given the 3 dimensional space the nodes traverse, we assume that the nodes are not constrained along a defined vector, although aircraft frequently operate along designated routes and at specified altitudes within the ATC system.

[1] Passenger and military jet aircraft typically fly at altitudes between 30,000 – 45,000 feet at speeds ranging from 450 – 500 mph. Turboprop aircraft normally operate at altitudes between 18,000 – 28,000 feet at speeds in the range 250 – 320 mph. Smaller general aviation aircraft are predominately powered by reciprocating engines, usually operating at altitudes below 18,000 feet and at speeds below 230 mph.

The network model is based on single-hop unidirectional broadcast links. Nodes in the MANET use a concept called beaconing to broadcast their position, velocity and direction in plaintext on recurring intervals of a few hundred milliseconds. We consider the ADS-B communications network to be a long range network, since it is designed to operate over wide coverage areas. The UHF frequencies utilized by the 1090 ES and UAT implementation of ADS-B are both line-of-sight (LOS), and are designed to operate at distances of 100 NM or more[2].

Although the ADS-B network has many similarities to wireless sensor networks, we assume that ADS-B devices have no energy limitations when actively participating as nodes in the network. Nodes that are equidistant from an ADS-B ground station are assumed to have the same signal strength with respect to that ground station. In addition, we assume that ADS-B ground station and aircraft avionics hardware have no significant computational constraints associated with sending and receiving messages on the network.

Another concern in some wireless sensor networks is the undetected physical capture of legitimate network nodes [11]. Since the aviation industry would consider it very undesirable to place any restrictions on ownership of general aviation aircraft, controlling legal access to legitimate ADS-B nodes would prove to be difficult if not impossible. As a result, the undetected physical capture of legitimate ADS-B nodes is a relatively low priority in the hierarchy of ADS-B vulnerabilities.

[2] Aviation distances are normally measured in nautical miles. One nautical mile is equivalent to 1.15078 statute miles.

As a final topic in describing the properties of the ADS-B network, we consider the overall network reliability. Throughout the development of the ADS-B system, network reliability has not been an important concern. As a consequence, the ADS-B protocol has no ability to mitigate collisions on the frequency channel. Due to the broadcast nature of the network model, there are no provisions in the protocol for handling lost packets. Although packet loss does not normally cause a problem for the sending and receiving of broadcast messages, there is a substantial amount of packet loss on the physical layer. According to [7] and [11], the mean packet error rate is 33%. This means that approximately 1/3 of the ADS-B messages currently exceed the 5-bit error correction limitation and this error rate will likely escalate as the ADS-B channel utilization rate increases due to the expected growth in air traffic density over the next several years.

Required Security Attributes

As discussed in the previous chapter, recent papers by Giannatto and Markowsky [5] and McCallie et al. [8] present several case studies which highlight the need for adding security to the ADS-B communications network broadcasts. In addition, work by Costin and Francillon [1], Magazu [2] , and Schäfer et al. [6] demonstrate the ease with which inexpensive and readily available hardware can exploit the vulnerabilities inherent in the existing ADS-B implementation.

The system performance standards for ADS-B are outlined in the Radio Technical Commission for Aeronautics (RTCA) documents DO-242A, DO-260B and DO-282B. None

of these documents make any mention of security as a part of the requirements specification. Therefore there was never any emphasis put on securing the protocol during its initial development. We can, however, use the network model defined in the previous section to identify the desired security attributes for potential ADS-B security schemes. An ideal and comprehensive ADS-B security solution will have the following qualities:

- *Compatibility* – The security scheme is compatible with the current ADS-B infrastructure and protocol, having minimal impact on current air traffic management operations.
- *Scalability* – The security solution is adaptable to increasing air traffic density and can accommodate anticipated growth in traffic volume. This implies that the solution must offer increased network reliability through robustness to packet loss.
- *Resistance to Signal Jamming and DoS* – The security solution will provide protection against malicious narrow band and pulse signal jamming attacks. The solution must also be secure against Denial of Service attacks.
- *Data Integrity* – The security scheme must provide assurance to the receiver that the data received has not intercepted and modified in any way by a third party.

- *Source Integrity* – The security approach must provide assurance to the receiver that the data received originated from the sender claiming to have sent the message.
- *Location Integrity* – The security scheme must provide assurance to the receiver that the message actually originated from the location claimed in the message position data.
- *Responsiveness* – Due to the very short communication timeframes in the MANET, the security solution must quickly detect and respond to incidents on the network.

Over the past few years there has been a growing body of work investigating possible approaches to ADS-B security, with progress in related fields such as wireless sensor networks and vehicular ad hoc networks (VANETs) providing researchers with ideas for developing security schemes applicable to ADS-B. Using the security requirements listed above as a guide, we will discuss and evaluate several current proposals for enhancing ADS-B security.

ADS-B Security Solutions Taxonomy

As noted above, there has been a substantial amount of recent research into providing ADS-B security, encompassing a variety of approaches. As shown in Figure 4, the proposed ADS-B security solutions can be organized into a taxonomy which groups the recent proposed security schemes into two separate and distinctive categories: **Secure Broadcast Authentication** and **Secure Location Verification**.

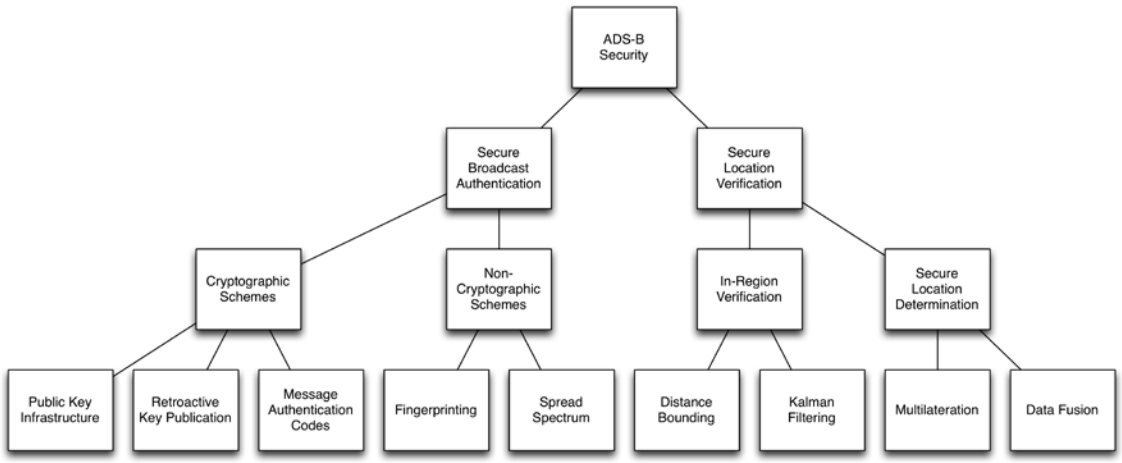

Figure 4. Taxonomy of ADS-B Security [7].

We categorize approaches to Secure Broadcast Authentication as those schemes which provide the receiver with verification that messages received actually originated from the claimed source and were not intercepted or modified en-route. These approaches are further subcategorized as **Cryptographic** and **Non-cryptographic** security schemes. In Chapter 5 we discuss both of these message integrity approaches.

Secure Location Verification schemes utilize a diverse group of non-cryptographic techniques to help verify the location claimed by a sender. Sastry et al. [12] distinguish between two different methods of secure location verification: **In-Region Verification** and **Secure Location Determination**. When a receiver employs in-region verification, various algorithms are used to analyze the available data and attempt to verify the plausibility of the sender's claimed location and intended vector. The receiver then either accepts or rejects the claim based on the probability that the sender is in the region claimed in the message.

In contrast, the secure location determination method attempts to discover the physical location of the sender as a means to cross-check the sender's claimed location. In this method, the receiver tries to compute the sender's actual location in 3-dimensional space and compare it to the location claimed in the message. In Chapter 6 we discuss secure location verification approaches of both the in-region verification and secure location determination types.

CHAPTER 5

SECURE BROADCAST AUTHENTICATION SOLUTIONS

In our description of the ADS-B network model, we noted that communications between nodes on the network are unidirectional broadcasts. Due to this broadcast architecture, potential security mechanisms must preserve the open nature of ADS-B so as not to restrict or encumber communications on the network. The lack of support for reliable data transfer and two-way communication between nodes in ADS-B makes message authentication more challenging than in a point-to-point communications network.

Security solutions in the Secure Broadcast Authentication category have been studied as a means for authenticating unidirectional broadcast messages. Recent work in securing MANETs and wireless sensor networks discuss both cryptographic and non-cryptographic solutions, and in the following sections we analyze their feasibility for providing security to the ADS-B communications system.

Cryptographic approaches include both symmetric and asymmetric mechanisms for message authentication[3]. Secure broadcast authentication schemes can be implemented either as a global mechanism on the network or designed so as to selectively respond to threats detected on the network. Such reactive authentication

[3] Symmetric-key algorithms utilize the same cryptographic keys for both encryption of plaintext and decryption of ciphertext, while asymmetric-key cryptographic algorithms require both a private and a public key.

could prove useful in reducing interference on the network by only requiring additional security at times when incidents seem more likely, minimizing additional computational and communicational overhead [7]. Non-cryptographic schemes focus on the physical layer, identifying solutions based on recognizing unique hardware or software characteristics of nodes on the network. In the following sections we discuss potential cryptographic and non-cryptographic security schemes.

Cryptographic Schemes

Cryptographic security schemes in wireless networks are an established means to secure communication that offer possible application to ADS-B. However, the open nature of the ADS-B architecture presents unique security challenges for cryptographic schemes, with a primary issue being the development of a suitable key distribution infrastructure.

Robinson et al. [13] describe the advantages and disadvantages to ad hoc and structured key distribution arrangements. An ad hoc approach to key distribution utilizes the preloading of trusted certificates into a node prior to the node joining the network. The trusted certificates would contain collections of public keys, and could be self-signed, signed by a local certificate authority (CA) or obtained from a third party distributor [13]. The validity of the certificates themselves cannot be verified by the node and must therefore be preloaded via a trusted mechanism. The principle advantage of ad hoc key distribution is its simplicity on relatively small networks. Having a limited number of certificates and corresponding private keys reduces the

probability of having a compromised key or an invalid certificate, assuming that a private key is not shared among multiple entities. The primary drawback to the ad hoc key distribution approach is that it does not scale well, since certificate management becomes much more challenging as the size of the network grows and the node density increases. Due to the lack of scalability of ad hoc key distribution arrangements, we will focus on cryptographic approaches that utilize structured key distribution.

Whether the cryptographic approach is symmetric or asymmetric, security techniques suitable for wireless sensor networks and MANETs cannot be simply retrofitted into the existing ADS-B communications system. This is due to several difficulties that the ADS-B network presents. For one, the ADS-B network is limited by the available UHF bandwidth on the 968 and 1090 MHz frequency channels and there are currently no plans for increasing the spectrum allocations. This creates an additional problem in that the number of nodes that the ADS-B system can support is limited by interference on the designated ADS-B frequency channels. Security solutions that extend the message length will result in increased interference and reduced operational capacity [9]. Also, any potential cryptographic schemes must be deployed globally, and therefore must be implemented jointly between several international aviation agencies.

Symmetric-key encryption utilizes algorithms to transform messages from plaintext to cyphertext and back using a secret cryptographic key shared by the sender and receiver. The encryption algorithms are designed to produce cyphertext that is computationally infeasible to decipher without the shared secret key. However, a

compact encryption scheme must be employed so as not create additional frequency congestion and interference problems.

One potential compact encryption solution is Format Preserving Encryption (FPE). FPE is a symmetric-key encryption algorithm that creates a cyphertext that is the same length as the original plaintext message, which means the encrypted messages would not add any additional communications load on the ADS-B channel. An alternative compact symmetric-key encryption solution for minimizing additional congestion on the ADS-B channel is to utilize a standard encryption algorithm in output feedback mode[4] with a block size that fits within the ADS-B message length restriction.

The primary drawback to symmetric-key encryption approaches to ADS-B is the problem of key management. In order for the ADS-B unidirectional broadcasts to be received and deciphered, all nodes on the network need access to the secret key. The problem is that anyone with knowledge of the secret shared key can generate valid messages, so a single secret key leak will compromise the entire security system. Since the ADS-B network environment is inherently untrustworthy and the open nature of the network requires all nodes to have access to the secret key, symmetric-key encryption schemes are an impractical approach to securing the current ADS-B implementation.

[4] Output feedback (OFB) is a mode of operation for a block cipher that permits encryption of differing block sizes, but the output of the encryption block function is the feedback (instead of the ciphertext). The XOR value of each plaintext block is created independently of both the plaintext and ciphertext.

Public Key Infrastructure

A structured key distribution solution called a Public Key Infrastructure (PKI) is a scalable approach to cryptographic key management. PKI makes use of an asymmetric-key encryption scheme, where each node on the network has a public-private key pair bound to a unique identity by a certificate authority. While less computationally efficient than symmetric-key encryption, asymmetric-key techniques have the advantage that a node cannot forge a message on the network. The unique public-private key pair guarantees that only nodes whose identities have been verified by the CA can communicate over the network. This means that if a node's private key is compromised, the CA need only revoke a single key pair, as the key pairs of all the other network nodes remain valid. In an asymmetric-key encryption scheme, nodes encrypt message with the intended recipient's public key using a standard asymmetric encryption algorithm. The receiving node then decrypts the message with its private key. Data integrity is ensured since only the sender's intended recipient can decrypt the message.

When considered as a security solution for ADS-B, asymmetric-key encryption has two major drawbacks. The first issue is that current asymmetric-key schemes have no compact encryption implementations, and would result in an increase of the transmitted ADS-B message length. The second problem is that unique encrypted ADS-B messages would be required for each recipient. To maintain a fully-connected network of n nodes would necessitate ($n^2 - n$) unique broadcasts rather than n in the current

system [9], which obviously does not scale well as the size of the network increases.

As a possible answer to these two drawbacks, Costin et al. [1] have suggested what they term a "lightweight" PKI solution. In the lightweight PKI approach, node *A* transmits its digital signature[5] over *n* messages, so that after every *n* messages, the surrounding nodes have received *A*'s digital signature. The recipients keep the messages until the entire digital signature has been transmitted and they can authenticate the buffered messages. The authors suggest that the PKI key distribution necessary for this scheme could be done during an aircraft's scheduled maintenance cycle [7].

As described by Zhang et al. [14] and outlined in [7], there are several obstacles in applying a full cryptographic solution to ADS-B that cannot be easily resolved. First, the open nature of ADS-B is widely seen as a desirable feature of the network. A cryptographic system intentionally obstructs public broadcast communication. Second, key exchange is notoriously difficult in ad hoc networks, which are by definition without a centralized institution. The dynamic nature of the network results in too much overhead in both the number and the size of messages. Third, any encryption scheme will immediately break compatibility with the existing infrastructure. For these reasons, it appears that a traditional, fully-cryptographic approach to securing ADS-B is not feasible.

[5] Digital signature algorithms take a message and a sender's private key as input and return a digital signature unique to the input. Upon receipt of a message-signature pair, the receiver can apply a verification algorithm to authenticate the signed message using the sender's public key.

Retroactive Key Publication

A security scheme called Timed Efficient Stream Loss-Tolerant Authentication (TESLA) is a variation on traditional asymmetric cryptography that has been proposed for use on broadcast networks [15], [16]. With TESLA, senders retroactively publish their keys which are then used by receivers to authenticate the broadcast messages. A broadcasting node produces an encrypted message authentication code (MAC) which is included with every message. After a designated time interval or number of messages, the key to decrypt the sender's MAC is published. Listening receivers who have buffered the sender's previous messages can then decrypt the messages that were broadcast. When applied to ADS-B, this technique imposes a time delay on the broadcast due to the need to buffer messages, but it provides integrity and continuity of messages sent over the network.

The TESLA protocol is loss-tolerant and scalable, capable of providing efficient broadcast authentication over networks consisting of a large number of nodes. µTESLA is an adaption of the TESLA protocol designed for use on wireless sensor networks. The µTESLA protocol requires nodes in the network to be loosely time synchronized, with each node having an upper bound on the maximum clock synchronization error. According to Perrig et al. [17], the µTESLA adaptation addresses several inadequacies of TELSA in wireless sensor networks:

- TESLA uses digital signatures for initial packet authentication, which are too computationally expensive for use in sensor nodes. The µTESLA protocol utilizes symmetric-key mechanisms.
- TESLA discloses a key with each message, generating too many messages on the network. In contrast, µTESLA releases the key once per time interval.
- TELSA stores one-way key chains for all nodes, which is expensive. The µTESLA protocol restricts the number of authenticated nodes.

As discussed earlier, asymmetric encryption schemes have high computation and communication overhead, which limit their usefulness as security approaches on the bandwidth-constrained ADS-B network. The µTESLA protocol overcomes this problem by employing asymmetric-key encryption through a delayed disclosure of symmetric keys, which results in an efficient broadcast authentication scheme. When one considers the bandwidth and interference limitations on the ADS-B frequency channel, the µTESLA design adaptations identify this protocol as a viable scheme for providing security in ADS-B.

To send an authenticated message, a sender computes a MAC on the message using a key that is secret at that point in time. When a recipient gets a message it uses its loosely synchronized clock, a upper bound on clock synchronization error and the time schedule at which keys are disclosed to verify that the corresponding verification key has not yet been disclosed by the sender. If the receiver determines that the key for

the message has not yet been disclosed, it is buffered for future authentication. At the scheduled time of key disclosure, the sending node broadcasts the verification key to all receivers. Once a recipient gets the disclosed key from the sender, it can readily verify the correctness of the key and authenticate the message stored in its buffer.

µTESLA use one-way key chains that are developed from the MAC included in the messages. Each MAC is used to generate a key in the key chain with a one-way function F. In order to generate the one-way key chain, a sender randomly chooses the last key K_n and repeatedly applies the function F to compute all the other keys: $K_i = F(K_{i+1})$, where $0 \leq i \leq n-1$. This means that every secret key K_i, where $i > 0$ is used for sending in the i^{th} interval and disclosed to the network after a scheduled time period t. Instead of adding a disclosed key to each data packet, the key disclosure is independent from the broadcast messages, and is tied to time intervals.

Part of the attractiveness of µTESLA is its ability to tolerate lost messages, and Figure 5 shows an example of how µTESLA copes with packet loss on the network. Each key K_i of the key chain corresponds to a time interval t_i, with all messages sent within time interval t_i authenticated with key K_i. In this example, the scheduled disclosure time period is 2. The example assumes that the receiving node is loosely time synchronized and that key K_0 has been previously authenticated on the network.

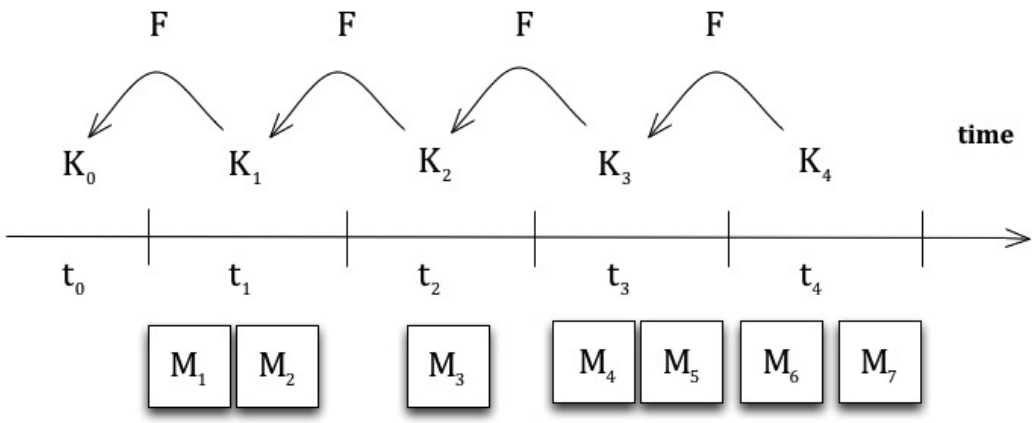

Figure 5. Example of µTESLA time-released key chain for source authentication [17].

In interval t_1 messages M_1 and M_2 are sent and contain a MAC created with key K_1. Message M_3 is sent in interval t_2 and contains a MAC generated using key K_2. At this point, the recipient cannot authenticate any of the buffered messages, as key K_1 has not yet been disclosed by the sender. Continuing with the example, assume that in interval t_3 messages M_4 and M_5 are lost. Further, let us assume that message M_6 disclosing key K_1 is also lost, so that the recipient is still unable to authenticate M_1, M_2 or M_3. In interval t_4 the sender broadcasts key K_2 , which the receiving node authenticates by verifying $K_0 = F\big(F(K_{2)}\big)$, and can determine the missing key since $K_1 = F(K_2)$. Using the disclosed keys, the recipient can authenticate messages M_1 and M_2 with K_1 , and M_3 with K_2 [17].

The µTESLA protocol is attractive as a security solution for ADS-B because it preserves the open nature of the broadcast network while avoiding a complex PKI infrastructure to ensure a sender's continuity. However, there are two obstacles to

applying µTESLA to ADS-B. The primary issue is that, while sufficiently good time synchronization could be provided via GPS, it would require modification to the protocol to accommodate the GPS timestamp field. The second problem is that in order for µTESLA to be used for verifying the identity of a network node, it needs to be reinitialized which leaves it susceptible to memory- based DoS attacks. In spite of these drawbacks, µTESLA is a promising security scheme for integrating into ADS-B.

Aircraft Address Message Authentication Code

The cryptographic solutions PKI and µTESLA both have shortcomings in that they require modifications to the current ADS-B protocol. In this section, we discuss a partial ADS-B security solution that focuses on establishing message source integrity rather than ensuring data integrity. The purpose here is to demonstrate a compatible security scheme that will mitigate threats posed by message injection and modification attacks, which are among the most critical vulnerabilities in the current ADS-B implementation.

The Aircraft Address Message Authentication Code (AA-MAC) security solution utilizes a standard hash algorithm such as MD5 or SHA and a secret authentication key to perform message integrity. The AA-MAC message source integrity scheme would require a slight modification to the existing protocol in that it would replace the current Aircraft Address (AA) field with the MAC, but the ADS-B message is otherwise unchanged, as shown in Figure 6.

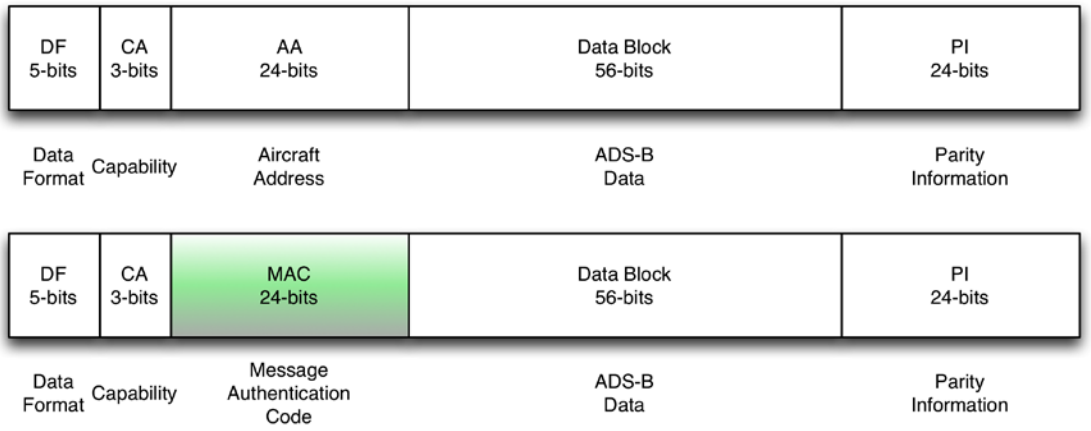

Figure 6. ADS-B message 24-bit PI field replaced by 24-bit MAC.

In the current ADS-B protocol, each aircraft is assigned a unique 24-bit Aircraft Address that is good for the life of the transponder equipment. The AA-MAC approach proposes a different aircraft identification strategy, assigning a unique identifier to each aircraft that is good for the duration of a particular flight. As with PKI cryptographic approaches, the distribution of the secret key presents challenges for AA-MAC. Several secret key distribution strategies have been proposed for PKI, such as distributing keys to all aircraft enclosed in tamper-proof hardware, utilizing an out-of-channel solution such as a separate dedicated frequency or distributing keys on a per-flight basis. Since MAC requires just one key which is used to uniquely identify a sender on the network, the simplest approach would be to distribute the secret key only when an aircraft intends to enter the air traffic control system and ADS-B network.

Every aircraft in the ATC en-route structure needs to file a flight plan prior to flight. The flight plan includes information about the equipment capabilities of the

aircraft, its intended route of flight, the expected duration of flight, the amount of fuel on board and other details that ATC needs to know for contingency planning. The flight crew could generate a secret key as part of their normal preflight procedure, and simply pass that key along to ATC with their flight plan. This would allow the secret keys to be continually updated, with a compromised key having minimal impact on the overall system.

A complication with this approach is that most common hash algorithms generate message authentication codes of 128-bits or more, which means the generated MAC itself is longer than the entire 112-bit 1090ES message. The generated hash needs to be shortened in order to meet the size limitation of the existing protocol. To accomplish this, a sender could compute the MAC using the secret authentication key, and then sequentially XOR the hash in 24-bit blocks (using 0 for padding) to produce a 24-bit MAC. This would then be inserted into the ADS-B message, using the 24-bit space allocated to the AA field.

This proposal has the potential to offer source integrity to ADS-B by establishing the identity of the sender; however there are several issues with this scheme that need to be addressed. One area of concern is the amount of source integrity afforded using a message authentication code of just 24-bits, yielding approximately 16.8 million different possible codes. Given that the ADS-B message it being broadcast in plaintext, the 24-bit size does not present a formidable computational challenge in forging the secret key. However, due to the mobility and speed of the nodes, the short duration

communications on the ADS-B network make the task of determining the secret key and forging messages far more difficult for an attacker than in a static a point-to-point network.

Additionally, since most hash functions produce MAC lengths of 128-bits or longer, the shortened MAC length increases the potential for collisions between duplicate MACs calculated for distinctly different messages. Appendix A contains a test program and supporting functions that were used to test for possible message collisions, where distinctly different messages hash to the same MAC. In the test, we randomly altered bits in an ADS-B message, and then computed a 24-bit MAC for both the original an altered messages using an MD5 hash function. The hashes were then reduced to 24-bit MACs and compared. In 10 different tests of 100 million iterations each, there were no collisions of duplicate MACs detected. While certainly not a comprehensive test it demonstrates that even with just a 24-bit hash, duplicate MACs will be a very rare occurrence.

The AA-MAC approach as we have described is limited to providing source integrity for air-to-ground communications. The system does not have a mechanism for establishing source integrity between aircraft. Combining AA-MAC with secure location verification approaches we discuss in Chapter 6 may offer a more comprehensive security scheme.

While AA-MAC does not provide data integrity, it is highly compatible with the existing 1090ES protocol and can be implemented at low cost relative to other security proposals, offering a feasible partial security solution for ADS-B.

Non-Cryptographic Schemes

As we have seen, cryptographic security schemes are difficult to implement in a way that is compatible with the existing infrastructure, primarily due to the problem of key distribution and management. Non-cryptographic approaches to network security avoid the challenge of key management and instead involve either some form of fingerprinting on the physical layer, or a frequency modulation scheme such as spread spectrum.

Physical Layer

Schemes such as fingerprinting encompass various methods for authentication and identification, either based on hardware or software imperfections or characteristics of the frequency channel which are hard to replicate. Regardless of the method employed, the goal is to detect and respond to suspicious activity in a network. Identifying signatures for legitimate nodes on the network provides data useful for the implementation of systems to detect network intrusions.

Software-Based Fingerprinting schemes attempt to isolate distinct characteristics of the software operating on network equipment. The development teams for different network equipment manufacturers often take widely varied paths when implementing software on a given device. These differences can be cataloged and later exploited to

tell apart dissimilar network devices, and can be used to verify their continuity up to a certain degree [7].

Hardware-Based Fingerprinting approaches seek to identify and catalog unique network hardware differences. Some of these differences can be used for radiometric fingerprinting, which takes advantage of differences in the modulation of a radio signal to catalog unique device signatures. Clock skew is another identifiable hardware feature that can be used to establish uniqueness between wireless devices. Since no two clocks are perfectly synchronized, time difference can be used to create signatures and enable identification.

The biggest obstacle to hardware or software fingerprinting in ADS-B is the difficultly in putting together a meaningful catalog for a fleet of similar aircraft. Commercial and military aviation fleets typically consist of hundreds of aircraft fitted with very similar or identical hardware, making them nearly impossible to differentiate. The similarity between fleet aircraft has the additional impediment that they are easier for a potential attacker to study and copy [7]. Hardware solutions such as clock skew are difficult to utilize in the current protocol as they would require timestamps included in ADS-B messages. Also, it is possible for an attacker to eavesdrop on the communication and mimic the appropriate clock skew.

A third category of fingerprinting is Channel/Location-Based Fingerprinting. This fingerprinting method tries to exploit natural characteristics of the communications channel. Various approaches utilizing received signal strength (RSS), channel impulse

response (CIR) and the carrier phase have shown that this can be a viable alternative to more traditional authentication and verification measures. They can be implemented relatively easily in wireless systems and can offer reasonable security without adding excessive overhead [7]. The drawback to Channel/Location-Based Fingerprinting is that it requires two-way communication, and thus is not compatible with the current ADS-B communications system.

Spread Spectrum

Another non-cryptographic solution used in securing RF communications is a method called spread spectrum. The technology is used in applications that require resistance to signal jamming and as a means of protecting wireless communications from passive listening. Spread spectrum methods have also proved beneficial as an aid in expanding the utilization of the available radio spectrum.

There are two approaches to spread spectrum frequency modulation: Frequency Hopping Spread Spectrum (FHSS) and Direct Sequence Spread Spectrum (DSSS). DSSS has the advantage of providing higher bandwidth capacity than FHSS, but it is a very sensitive to environmental factors. FHSS is a more robust technology than DSSS, with little susceptibility to interference. In addition, FHSS can accommodate a significantly higher number of simultaneously active systems in the same geographic region than DSSS systems. These characteristics make the FHSS technology much better suited to the ADS-B network.

Spread Spectrum utilizes a key (also called the code or sequence) attached to the communication channel. The way in which the code is attached to the communication channel expands the signal bandwidth by several orders of magnitude and determines whether the spread-spectrum technique is Frequency Hopping or Direct Sequence. The baseband signal is intentionally spread over a larger bandwidth by injecting a higher frequency signal. As a result, energy used in transmitting the signal is spread over a wider bandwidth, and appears as noise. The ratio (in dB) between the spread baseband and the original signal is called processing gain, with typical spread spectrum processing gains ranging from 10dB to 60dB [18].

When spread spectrum is applied to the communications channel the effect is to diffuse the information into a larger bandwidth. This diffusion process is what provides security and protects the channel from jamming and eavesdropping attacks. The receiving node can remove the spread-spectrum code in a process called de-spreading. The de-spreading operation reconstitutes the information into its original bandwidth so that the data can be retrieved.

There are two primary drawbacks to applying spread spectrum to the ADS-B channel. The first is that the spreading code must be known in advance at both ends of the transmission channel, and thus spread spectrum has the same key distribution and management issues that encumber cryptographic techniques. The second issue for ADS-B is that spread spectrum is incompatible with the current ADS-B infrastructure, requiring substantial changes that would be costly to implement.

Due to the need for extensive infrastructure modifications, non-cryptographic security schemes such as channel/location-based fingerprinting and spread spectrum are not feasible security solutions for the current ADS-B infrastructure. However, channel/location-based fingerprinting does provide a non-cryptographic means of authentication while spread spectrum offers protection against signal jamming attacks. Based on these desirable attributes, development of a post-NextGen ATC system should explore ways to incorporate these non-cryptographic security solutions.

CHAPTER 6

SECURE LOCATION VERIFICATION SOLUTIONS

As an alternative to securing the communication channel of ADS-B, the concept of Secure Location Verification is to substantiate the authenticity of location claims made by ADS-B network participants. This approach is inherently different from establishing the integrity of broadcast sources and messages. As described in [12], In-Region Verification and Secure Location Determination are two different methods of location verification, but the underlying principle of both is to confirm a node's position within the network. The goal of secure location verification schemes is to provide a means of cross-checking location claims made by network participants. Since secure location verification creates supplementary position data, these approaches have the additional advantage of offering redundancy to the current system. This additional data can be merged with ADS- B and radar information, providing a fallback system in case of failure of the primary surveillance system [7].

In-Region Verification

There is a distinction between secure source location verification methods that attempt to precisely identify the location of a network participant, and those that attempt to determine the plausibility that a sender is in the region claimed in a message. In-Region Verification schemes attempt to do the latter, employing

algorithms that utilize estimation methods to determine the probability that a sender's claim is true. We discuss Distance Bounding and Kalman Filtering as two possible In-Region Verification solutions with potential application to ADS-B.

Distance Bounding

Distance bounding is a location verification method employed in wireless networks to localize other network nodes. The basis of distance bounding protocols is built on the fact that electromagnetic waves travel at roughly the speed of light, but never faster. The concept behind distance bounding is to provide a means wherein a location claimed by prover *P* is challenged by a verifier *V* to demonstrate that *P* is within a certain physical distance of *V*, as shown in Figure 7.

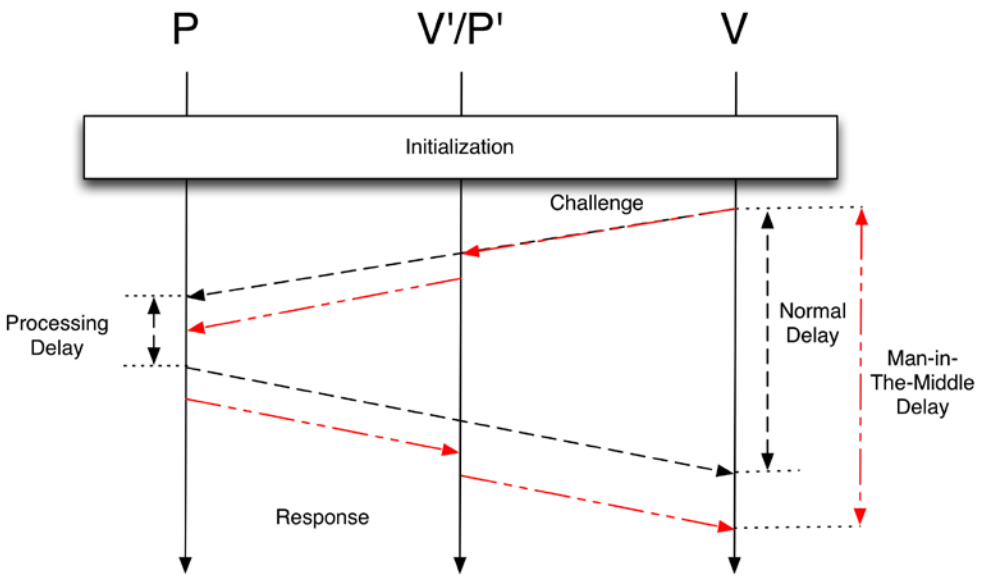

Figure 7. Principle of distance bounding protocols [7].

The verifier *V* sends a challenge message indicated by the dashed black arrows in

Figure 7 to the prover *P* who then, after processing the challenge message, sends its response. As indicated by the alternating solid and double-dash arrows in the figure, a man in the middle (*V'*/*P'*) can only increase the apparent distance by adding further processing delay. This enables *V* to compute a distance based on the time between the *V*'s challenge and the corresponding response by *P*. The distance computed by *V* serves as an upper-distance bound between *V* and *P*, and can be used to check the truth of *P*'s claimed location. When applied to the air traffic surveillance model, the distance-bounding results of multiple ground stations can be combined to establish the probability that a sender is actually in the region claimed in its message.

There are several potential challenges to implementing distance bounding into ADS-B. The first issue is that distance bounding schemes have been used primarily for close-range indoor wireless communications, and have not been successfully tested over the long distances and with the high node velocities present in the air traffic control system. Another issue is that there are various practical attacks on distance bounding schemes given in the literature, among them a number of relay attacks and distance hijacking attacks, so further research into mitigating the effectiveness of the these threats is required. Perhaps the most problematic issue facing distance bounding schemes is the fact that it requires a challenge-response protocol, which renders them incompatible with the current ADS-B implementation.

Kalman Filtering for Intent Verification

Kalman filtering is a technique used to filter observations from a noisy data series by providing estimates for the future state of values in the underlying system. The theory behind Kalman filtering requires the observed system to be a linear data series and the underlying input variables to follow a normal distribution. The algorithm is recursive and can efficiently update its estimation values in real time, without having to save more than the last system state. The Kalman filtering procedure comprises three distinct steps; prediction, observation and update. In the first step, the current system state variables are predicted along with a probability estimate of associated uncertainties. The predicted system state values are based on an estimation of the current system state, observations of the state transitions in the system and known system input control variables. The predicted uncertainties are probability estimations based on observations of the state transitions in the system, a system data covariance estimate and an estimate of system process error.

In the observation step, the system adjusts its estimation values by measuring the residual, which is the discrepancy between the predicted values from the previous step and the currently observed system state values. The observed system values are also used to calculate a residual covariance, based on the probabilities from the previous step and an estimate of system observation measurement error. Finally, the previously obtained estimates are updated with a weighted average, and the estimates with higher probabilities are assigned higher weights.

This continuous process of prediction, observation and update is a form of feedback control, where the filter estimates the process state at a point in time, and then obtains feedback in the form of noisy measurements. The update step provides the feedback, which incorporates the new observation into the existing estimate to obtain an improved estimate [19]. Thus, a system employing Kalman filtering is constantly updating itself with the most recent observations and revising the values used to compute its estimates.

Figure 8 shows an example of a Kalman filter applied to a single-variable system. The figure was generated using a linear Kalman filter implemented in Python, which is listed in Appendix B.

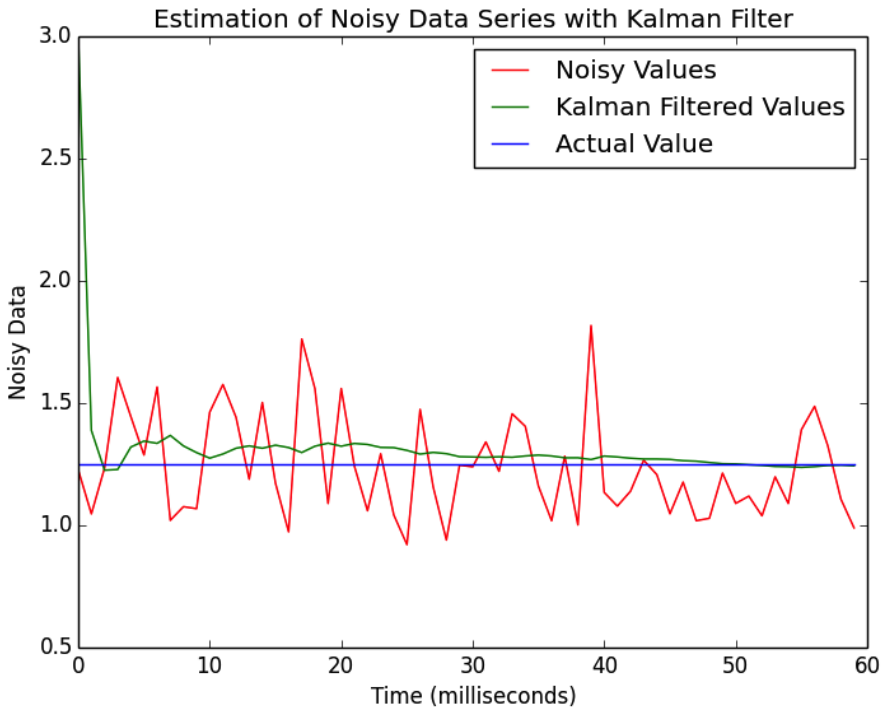

Figure 8. Single-variable Kalman filtering example.

In the example, the noisy values are randomly generated from a Gaussian distribution with a mean of 1.25 and standard deviation of 0.25. The figure provides a demonstration of the estimation refinement process that takes place in the feedback control loop. As the figure shows, the Kalman filter quickly develops an estimate of the actual value. The estimation values provided by the Kalman filter continually improve in accuracy, so that by 30 milliseconds, it has a very close estimation of the actual value of 1.25 and it maintains a reasonably precise estimation through the remainder of the time period. When employed as means of intent verification in the air traffic management system, a multi-variable Kalman filter would be utilized, but the underlying estimation refinement process for each input variable would be the same as shown in the single-variable example.

Kalman filtering is currently used in airport ground control systems to help prevent runway incursion incidents and minimize bottlenecks on airport taxiways. This is accomplished by filtering and verifying data reported by aircraft equipped with ADS-B and conducting plausibility checks on these observations. Krozel et al. [20] propose a multi-variable Kalman filtering solution that can be used to verify the intent of an aircraft by identifying correlation functions that can in turn be used to evaluate relationships between actual aircraft motions and the intended vector information sent in the ADS-B message. The proposal uses data obtained from ATC controller directives regarding heading and altitude to determine the target's geometric conformance. The system then evaluates the actual aircraft heading and altitude and compares it to the

target's broadcast intentions to determine intent conformance. The two conformances are then analyzed to develop a plausible intent model in terms of the estimated horizontal and vertical paths, as well as the anticipated velocity of the target.

Kalman filtering has two principal weaknesses that leave it vulnerable to possible exploitation. One is that Kalman filters can be tricked by what is termed a frog boiling attack. In this exploit, an adversary jams the message transmission legitimate of a legitimate node while continuously transmitting a slightly modified bogus position message. If the adversary transmits the false data slowly enough, the Kalman filter will see this injected data as a valid trajectory change. A second potential weakness of the scheme is that it opens up more DoS attack possibilities, since the Kalman filtering process requires increased computational complexity at every node in the network [7]. In spite of these weaknesses, Kalman filtering is very compatible with the existing ADS-B infrastructure, offering a highly scalable and relatively low cost means of adding security to the current ADS-B communications system.

Secure Location Determination

In contrast to In-Region Verification schemes which attempt to determine the plausibility of a sender's claimed location, the general principle behind Secure Location Determination is to identify the precise position of a node on the network as a cross-check of the location claimed by the sender. Because Secure Location Determination solutions are used to verify a sender's actual location, they are redundant surveillance systems and require an infrastructure that is independent of the ADS-B communications

system. We will discuss two potential Secure Location Determination approaches for ADS-B, Multilateration and Data Fusion.

Multilateration

Multilateration (MLAT) is a low-cost location determination technology that is used for airport surface movement as well as for terminal and en route traffic surveillance. MLAT provides excellent performance under a variety of conditions, and is especially useful in providing surveillance in remote geographic areas due to its less frequent maintenance requirements than radar systems. MLAT is a completely independent surveillance system, and unlike ADS-B, does not require any change or modification to existing aircraft avionics or communications systems.

A multilateration system is based on the time difference of arrival (TDOA) principle. The system requires multiple antennas in separate locations that receive the same signal, but at different times due to TDOA. A typical system consists of four or more target tracking antennas and a target processing unit as shown in Figure 9.

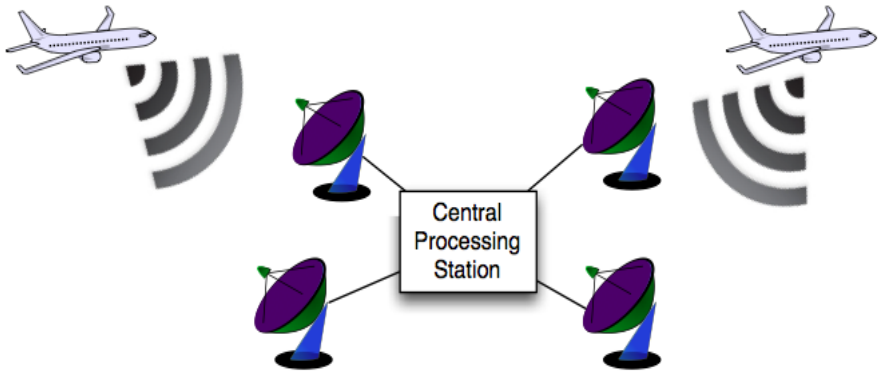

Figure 9. Basic MLAT architecture [7].

The target processing unit calculates a target's position based on the TDOA of the signal as measured at the tracking antennas. Since the target processing unit knows the exact location of each tracking antenna, the measured TDOA at two receivers can be used to form a hyperboloid in the region where the target is located. When the TDOA of four or more receiving stations are combined, the result is the intersection of three or more hyperboloids. This intersection allows the target processing unit to identify the target's position in 3-dimensional space, as shown by the red point in Figure 10.

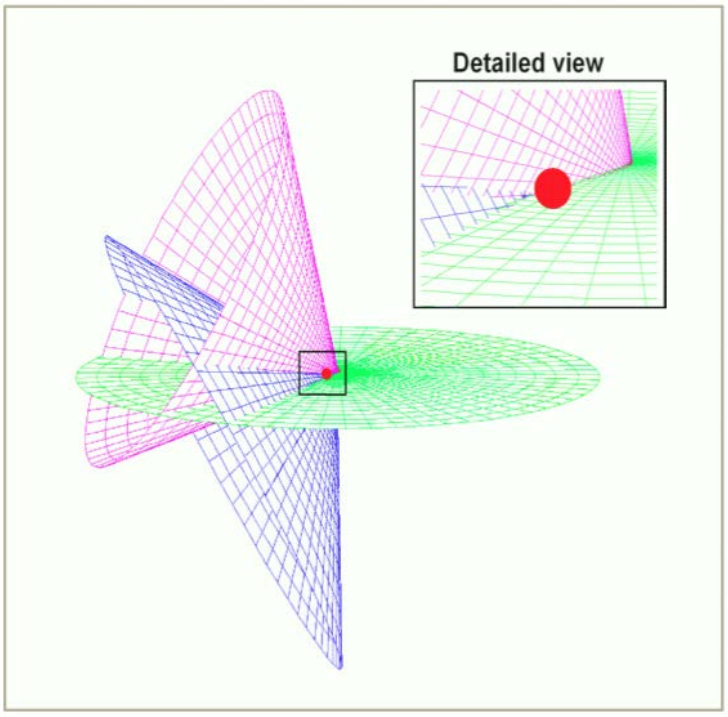

Figure 10. Intersection of Three TDOA Hyperboloids [21].

MLAT is currently used as a ground surveillance technology at various airports, but there have been recent studies of Wide Area Multilateration (WAM) for application in airborne MANETs. Recent work with WAM has shown that it is possible to obtain

roughly ±30 meter accuracy at a distance of 90 NM from the central station. This compares favorably with the ±20 meter accuracy obtained from ADS-B via its GPS data. WAM offers target position accuracy comparable to ADS-B at distance up to 100 NM, but beyond that distance WAM precision begins to degrade rapidly [7].

In order to achieve sufficient system accuracy, individual tracking antennas in a purely hyperbolic system should be as far apart as possible. However, coverage volume and geographic considerations do not always allow for optimum ground station placement. To solve this problem, Xu et al. [21] discuss an elliptic-hyperbolic MLAT system that has been the subject of recent research. In this system, an ellipsoid is created by a sender and a receiver, using the known total time between a Mode S interrogation and its reply. The ellipse is formed from the reply time sum of arrivals (TSOA) as shown in Figure 11, and tends to intercept the juncture of the MLAT hyperboloids.

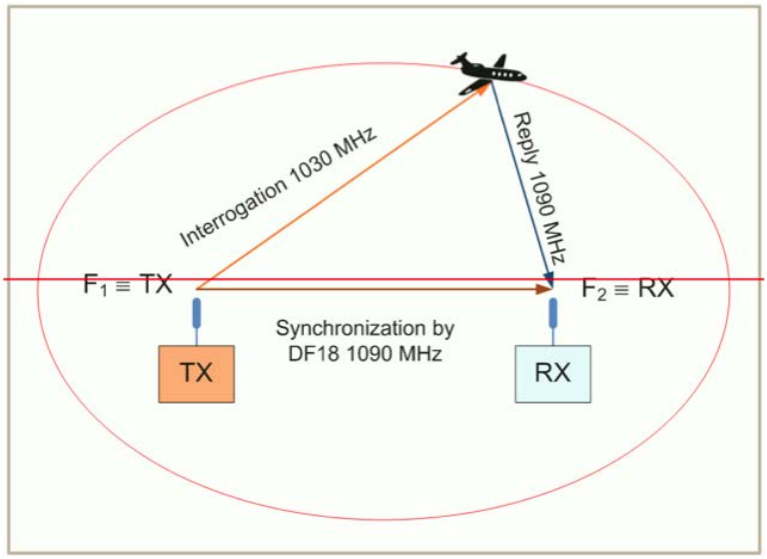

Figure 11. Construction of TSOA Ellipse [21].

A weakness of purely hyperbolic MLAT systems is their lack of precision in estimating target altitude. The system is capable of providing precise 2-dimensional latitude and longitude estimates, but has difficulty in accurately determining target altitude with its ground-based antenna due to the angle of intersection between the hyperbolas. The elliptical-hyperbolic MLAT implementation significantly reduces altitude estimation errors and can yield more accurate 3-dimensional position estimates than purely hyperbolic MLAT systems.

In spite of the performance potential and low cost of the system, there are some unsolved problems to applying MLAT as a secure location determination solution. The first issue is that the system is susceptible to RF interference phenomena which result in multipath propagation, where the signal reaches the antenna via different paths. This can distort the TDOA information resulting in an erroneous calculation of target position.

Another technical issue affecting MLAT is the requirement for the target signal to be correctly detected at multiple receiving stations in order for the target processing unit to determine an accurate position. The large number of required ground stations increases the probability of an equipment failure in the system, which would degrade the central station's ability to provide accurate target position estimates. As a final point, WAM systems may have difficulty scaling to meet increasing air traffic density. Since MLAT relies on multiple ground-based antennas, the current WAM implementation proposals may reach system capacity in certain regions where geographic characteristics prevent the installation of additional tracking antennas.

Data Fusion

Data fusion is a recognized method for aggregating data from different sources, with the goal of producing information that is more valuable to the end user than the original individual data sets. Data fusion is a multilayered process that uses associations and correlations in data from multiple sources to create estimates which are combined into a single data set. The concept can be employed using a variety of approaches

including statistical analysis, probabilistic modelling, fuzzy logic and machine learning. The type of data fusion approach used depends on the requirements of the application, the type of data being analyzed and the desired reliability of the result [22].

When considered as an ADS-B security scheme, the literature proposes the use of estimation algorithms to check positional data obtained from within the ADS-B communications network against data coming in from independent surveillance sources such as PSR, SSR and MLAT (see Figure 12).

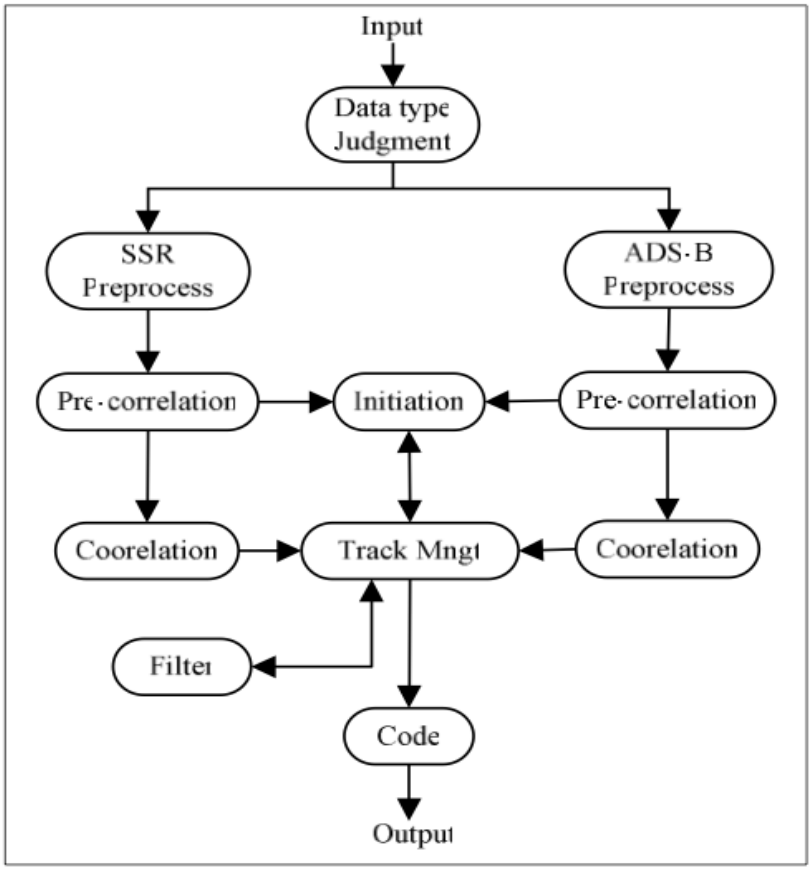

Figure 12. ADS-B/SSR Fusion Model [23].

These estimation and verification methods can provide a means of error and threat detection, by determining if some of the involved system data is outside normal parameters. This allows automated procedures to be developed which would permit quick problem identification and reaction within the air traffic management system. The central concept to utilizing data fusion as a secure location verification paradigm is establishing the trustworthiness of the data, thus determining if it has been subject to error or malicious modification. The data trust-worthiness can be calculated by analyzing the data associations and correlations using fusion algorithms, which aim to expose anomalies in received information and thus to enable the automated detection of threats to the system [7].

Yong et al. [23] discuss some challenges facing the data fusion of ADS-B and radar surveillance information. One issue is that the two surveillance approaches utilize different coordinate systems. Since ADS-B position data is derived from GPS, it uses the WGS-84 coordinate system, while PSR and SSR use polar coordinates. The authors suggest the positional data be transformed to Cartesian coordinates. Another complication is that the system needs to be able to handle time calibration differences between the information sources. The data provided by ADS-B is obtained non-synchronously with the radar or MLAT data it is being compared against. Therefore, a time bias needs be calibrated prior to applying the data fusion algorithm.

Since many of the required components for data fusion are already contained within the existing surveillance infrastructure, integrating it as a secure location verification solution is relatively straightforward. The advantages of extending data fusion techniques to secure location verification with ADS-B are its compatibility with legacy systems and the fact that the ADS-B protocol is not affected by data fusion security schemes. The obvious drawback is the increased cost due to the requirement for multiple independent surveillance information sources.

CHAPTER 7

ANALYSIS OF SECURITY SOLUTIONS

After reviewing potential approaches to secure broadcast authentication and secure location verification, it is clear that there is no single optimal solution to securing ADS-B communications. Limitations in the existing ADS-B protocol, congestion on the 1090 MHz channel and the need for compatibility with existing communications hardware present challenges to finding viable security solutions and render many proposals impractical.

Cost-Effective Solutions

A concern that is notably absent in the recent literature on ADS-B security solutions is the substantial cost to the aviation industry of installing the required avionics equipment to support the ADS-B surveillance system. Every military, commercial and general aviation aircraft operating in the ATC system will require additional avionics equipment or modifications to existing equipment in order to support ADS-B communications. In addition, pilots and air traffic controllers need to be trained on the new equipment. According to the airline industry, ADS-B equipment could cost airlines, including regional carriers, as much as 5 billion dollars [24] in order to meet the FAA mandated implementation deadline of January, 2020. The price tag for modifying the military and general aviation aircraft fleet is equally as staggering. It is unreasonable to expect the airline industry, the DoD and private aircraft owners to

invest billions of dollars in new avionics equipment only to have that equipment rendered obsolete by modifications to the ADS-B system to support additional security.

Therefore, any security measures employed in the ADS-B system must take cost and compatibility with the existing hardware and protocol into account. In our analysis of security solutions, we place a high degree of emphasis on compatibility of the various proposals with the current ADS-B implementation. Strohmeier et al. [7], develop an interesting tabular comparison of the capabilities, security features and feasibilities of various security approaches for use with ADS-B. We expand on the tabular comparison in [7] to create a ranking system for evaluating the most cost-effective and feasible solutions out of those currently being studied.

Evaluation of Scheme Implementation Considerations

Using the security scheme characteristics described in Chapter 5 and Chapter 6, we evaluate the various security approaches using three categories; *implementation complexity*, *type of security provided* and *message integrity features provided*. In Table 2, we evaluate the following security scheme properties:

- *Difficulty* - The overall complexity of implementing the approach. We categorize the scheme difficulty as follows:
    - *High* - Schemes that require regulatory changes such as the need for additional frequency bandwidth or major infrastructure modifications are considered difficult to implement. Schemes with High difficulty are assigned 0 points in our ranking system.

- *Moderate* – Schemes that require at least some changes to the existing infrastructure or modifications to the protocol are considered moderately difficult to implement. Schemes with Moderate difficulty are assigned 0.5 points in our ranking system.
- *Low* - Schemes that require no changes to infrastructure or protocol are considered to have a low implementation difficulty. Schemes with Low difficulty are assigned 1 point in our ranking system.

- *Cost* - The projected cost of required hardware and software changes. We categorize a schemes cost as follows:
  - *High* - Proposals that require new avionics and ground station hardware or major software design changes are considered costly. Schemes with High cost are assigned 0 points in our ranking system.
  - *Moderate* – Schemes that require minor hardware and/or software are considered moderately costly to implement. Schemes with Moderate cost are assigned 0.5 points in our ranking system.
  - *Low* - Schemes that require minimal changes to hardware or software are considered to have a low implementation cost. Schemes with Low cost are assigned 1 point in our ranking system.
- *Scalability* - Considers how well the proposed scheme can adapt to rising air traffic density. We categorize a schemes scalability as follows:
  - *High* - Highly accommodative schemes are considered scalable. Schemes with High scalability are assigned 1 point in our ranking system.

- *Moderate* – Schemes that will have some difficulty in handling increasing traffic density are considered moderately scalable. Schemes with Moderate scalability are assigned 0.5 points in our ranking system.
- *Low* - Schemes that will have substantial difficulty in accepting increasing traffic density are considered to have low implementation scalability. Schemes with Low scalability are assigned 0 points in our ranking system.

- *Compatibility* - Evaluates the proposed security solution based on its impact on current operations. Feasible schemes should not excessively impact current hardware and software standards. Our ranking system deducts 0 points for schemes that require no changes, 1 point for schemes that require changes to the existing infrastructure and an additional 1 point for schemes that require changes to the existing ADS-B protocol.

Table 2 lists the security schemes in descending order, from easiest to integrate into the current system to the most difficult. Based on our ranking system for Implementation Considerations, the maximum score for a security scheme would be 3, having a low implementation difficulty and cost, a high scalability and be completely compatible with the existing ADS-B communications system. From the rankings shown in Table 2, we can see that Kalman filtering, MAC and wide-area multilateration security schemes are the most compatible with the existing ADS-B infrastructure and protocol. All three approaches have a low implementation difficulty, have relatively low cost, are scalable and have a high degree of compatibility with the current system.

Type	Implementation Considerations				Overall Score
	Difficulty	Cost	Scalability	Compatibility	
Kalman Filtering	Low	Low	High	No additional messages needed. Separate software system.	
	1	1	1	0	3
Message Authentication Codes	Low	Low	High	Key distribution infrastructure, minor change to ADS-B protocol.	
	1	1	1	-1	2
Wide Area Multilateration	Low	Medium	Medium	Utilizes a separate hardware system. No change to existing ADS-B required.	
	1	0.5	0.5	0	2
Data Fusion	Low	High	Medium	No change in ADS-B required. Requires independent system(s).	
	1	0	0.5	0	1.5
µTESLA	Medium	Medium	High	Protocol requires a new message type for key publishing.	
	0.5	0.5	1	-1	1
Physical Layer Authentication	Medium	High	Medium	Requires additional hardware/software. No modifications to the ADS-B protocol.	
	0.5	0	0.5	-1	0
Distance Bounding	High	Medium	Low	New messages and protocol needed.	
	0	0.5	0	-1	-0.5
Spread Spectrum	High	High	Medium	Requires new hardware and a new physical layer. Requires modifications to the ADS-B protocol.	
	0	0	0.5	-2	-1.5
(Lightweight) PKI	High	High	Medium	Key distribution infrastructure and changes in protocol and message handling needed.	
	0	0	0.5	-2	-1.5

Table 2. Scheme Implementation Considerations

The most difficult security approaches to integrate are public key infrastructure and spread spectrum. Due to the need to modify the existing hardware and protocol, these schemes are the least desirable from an implementation complexity viewpoint.

Evaluation of Scheme Security Provided

We next compare the different security schemes by the type of security provided by the approach. In Table 3, we evaluate the following security scheme properties:

- *Injection / Modification* –Indicates whether or not the scheme offers protection against target injection and/or message modification attacks. Schemes that offer protection against these attacks are assigned 1 point in our ranking system.
- *Eavesdropping* – Specifies if the scheme offers protection against passive listening. Schemes that offer protection against these attacks are assigned 1 point in our ranking system.
- *Jamming* - Considers how well the proposed scheme can protect against signal jamming attacks. Schemes that offer protection against these attacks are assigned 1 point in our ranking system.
- *Denial of Service Mitigation* - Shows whether or not the proposed security solution offers protection against target injection and/or message modification attacks. Schemes that offer protection against these attacks are assigned 1 point in our ranking system.

Table 3 lists the security schemes in descending order based on the types of security protection offered by the approach. In our ranking system for Security Provided, the maximum score for a security scheme would be 4 if a scheme offered protection against all attack categories.

Type	Security Provided				Overall Score
	Injection / Modification	Eavesdropping	Jamming	DoS Mitigation	
Spread Spectrum	No	Yes	Yes	Yes	
	0	1	1	1	3
(Lightweight) PKI	Yes	Yes	No	Yes	
	1	1	0	1	3
Physical Layer Authentication	Yes	No	No	Yes	
	1	0	0	1	2
Data Fusion	Yes	No	No	Yes	
	1	0	0	1	2
Message Authentication Codes	Yes	No	No	No	
	1	0	0	0	1
µTESLA	Yes	No	No	No	
	1	0	0	0	1
Wide Area Multilateration	Yes	No	No	No	
	1	0	0	0	1
Distance Bounding	Yes	No	No	No	
	1	0	0	0	1
Kalman Filtering	Yes	No	No	No	
	1	0	0	0	1

Table 3. Scheme Security Provided

Most of the security schemes that are currently being explored focus on attacks of the message injection and modification type. There is currently not a great deal of interest in protection against passive listeners, even though eavesdropping is often the first step in developing more sophisticated and problematic attacks [7]. This is primarily

due to the difficulty in developing adequate protection against passive listening without resorting to a full cryptographic solution. Of the security approaches we have reviewed, only spread spectrum and PKI offer protection against passive listening. In addition, spread spectrum is the only security scheme that offers protection against signal jamming attacks. As a result these two schemes score highest in our ranking of Security Provided.

Evaluation of Scheme Message Integrity Provided

The next comparison table we construct ranks the security schemes by the type of message integrity offered. In Table 4, we evaluate the following security scheme properties:

- *Data Integrity* –Indicates whether or not the scheme ensures that the data is the same as has been provided by the sender and has not been modified by any third party. Schemes that offer this feature are assigned 1 point in our ranking system.
- *Source Integrity* – Specifies if the scheme can ensure that a message originates from the participant that claims to have sent it. Schemes that offer this feature are assigned 1 point in our ranking system.
- *Location Integrity* - Considers how well the proposed scheme can determine that a message originates from the location claimed in the message. Schemes that offer this feature are assigned 1 point in our ranking system.

Table 4 ranks the security schemes in descending order based on the level of message integrity that the approach adds to the communications network. In our ranking system for Message Integrity Provided, the maximum score for a security scheme would be 3 if a scheme offered data integrity, source integrity and location integrity for surveillance communications messages.

Type	Message Integrity Provided			Overall Score
	Data Integrity	Source Integrity	Location Integrity	
(Lightweight) PKI	Yes	Yes	Yes	
	1	1	1	3
Data Fusion	No	Yes	Yes	
	0	1	1	2
Physical Layer Authentication	No	Yes	No	
	0	1	0	1
Message Authentication Codes	No	Yes	No	
	0	1	0	1
µTESLA	No	Yes	No	
	0	1	0	1
Kalman Filtering	No	Yes	Yes	
	0	1	1	2
Distance Bounding	No	No	Yes	
	0	0	1	1
Spread Spectrum	No	No	No	
	0	0	0	0
Wide Area Multilateration	No	No	No	
	0	0	0	0

Table 4. Scheme Features Provided

As we can see from the comparison in Table 4, only a full cryptographic public key infrastructure can guarantee the integrity of received data [7]. Cryptographic security solutions such as PKI, whether implemented symmetrically or asymmetrically, are the only security schemes that offer protection in all three message integrity categories. All other security approaches offer only a partial solution.

Summary of ADS-B Security Schemes

Table 5 summarizes the scores from Table 2 - Table 4 and provides an overall ranking of the ADS-B security schemes discussed.

Type	Implementation Considerations Score	Security Provided Score	Message Integrity Score	Overall Score
Kalman Filtering	3	1	2	6
Data Fusion	1.5	2	2	5.5
Message Authentication Codes	2	1	1	4
(Lightweight) PKI	-1.5	2	3	3.5
µTESLA	1	1	1	3
Wide Area Multilateration	2	1	0	3
Physical Layer Authentication	0	2	1	3
Spread Spectrum	-1.5	3	0	1.5
Distance Bounding	-0.5	1	1	1.5

Table 5. Scheme Ranking Summary

To review the scoring system defined in Table 2 - Table 4, the maximum score for the Implementation Considerations of a security scheme is 3, the maximum score for Security Provided is 4 and the maximum score for Message Integrity is 3 for a maximum possible overall score of 10 in our ranking system. Based on the security scheme ranking criteria outlined above, the three most cost-effective and feasible security solutions are Kalman filtering, data fusion and message authentication codes.

As discussed in Chapter 6 and shown in Table 5, Kalman filtering for use in real time positional claim verification is among the easiest of the schemes to implement. Although Kalman filtering solutions provide limited security and message integrity, their low overall adverse impact on the existing ADS-B communications system make them a suitable security approach for integrating into the current surveillance system.

Data fusion schemes can be used to verify positional data obtained from within the ADS-B surveillance system against data acquired from other, independent surveillance sources such as PSR and SSR. As with Kalman filtering approaches, data fusion provides additional security to the ADS-B surveillance system while also having a high degree of compatibility with the current system. The obvious drawback to data fusion is the requirement to maintain redundant sources of surveillance data. The additional maintenance cost implies that the FAA will not achieve the cost reduction benefits hoped for as part of the NextGen deployment, but the cost will certainly be less than that of other proposed ADS-B security schemes.

As we demonstrated with AA-MAC in Chapter 5, message authentication codes can be implemented into the existing ADS-B message protocol with a minimal overall adverse impact on the existing system. A security scheme that employs MAC would provide protection against message injection and modification attacks, which are considered to be among the most critical of the vulnerabilities in the current ADS-B surveillance system. The nominal adverse impact on existing ADS-B communications make MAC security schemes worth incorporating into the current system.

Although spread spectrum and pure cryptographic solutions scored low in our ranking system, they are necessary schemes to consider in developing a post-NextGen air traffic management system. As noted earlier in this chapter, spread spectrum is the only security scheme that offers protection against signal jamming, while PKI is currently the only method for ensuring data integrity. Therefore, both of these security schemes should be considered essential security components in future air traffic management systems.

CHAPTER 8

FUTURE WORK & CONCLUSION

The FAA's NextGen upgrade was intended to increase the air transportation system capacity and safety while reducing its operational cost, but recent research demonstrates that potential vulnerabilities in the implementation of the ADS-B component of NextGen can be easily exploited with inexpensive and readily available equipment. The FAA began to develop NextGen and ADS-B at a time when network security was not a concern and the term "cybersecurity" did not exist. At that time, the system development effort was focused on reliability, accuracy, ATC system operational capacity, and range [10]. Since then, network attacks have become an everyday occurrence and the need for robust security measures are now a crucial network design consideration. In addition, the tragic events of September 11, 2001 exposed how vulnerable our global air transportation system is to those seeking to exploit inherent weaknesses in the system with nefarious intent.

The FAA projects a 48% increase in domestic commercial air travel between 2014 and 2034. According to the FAA's most recent Aerospace Forecast, U.S. commercial air carrier system enplanements are anticipated to increase from approximately 775 million in 2014 to over 1.150 billion by 2034 [25]. Based on this expected increase in air traffic, it is clear that aviation authorities urgently need to mitigate the security problems in NextGen. In addition to the expected increase in commercial air travel, as the FAA moves to develop rules for integrating Unmanned Aerial Vehicles (UAVs) into the air

transportation system the need to safely accommodate the resulting increase in traffic density becomes even more critically important.

As we have shown, there is no single comprehensive security solution to address the vulnerabilities in the current implementation of ADS-B. Given the decades-long timeframe required to develop, certify and deploy an air traffic management system and its substantial costs to the aviation industry, a complete overhaul of ADS-B is not a feasible consideration. Therefore, any solution to addressing the security shortcomings in ADS-B will be a compromise and partial answer to addressing the vulnerabilities in the system.

Viable ADS-B security solutions should seek to apply incremental changes to the current system with an emphasis on backwards compatibility. However, implementation expense and complexity of deployment should not be the only factors taken into consideration; not having a security solution might prove to be far more costly in the long run [7]. In view of this dichotomy, it seems logical to pursue two different future research directions; one focused on addressing some of the weaknesses in the current system and the other focused on developing a successor to ADS-B using the existing system as a case study for future work.

NextGen Future Research

In our analysis of cost-effective solutions to establishing secure broadcast authentication, we discuss AA-MAC as a security enhancement that has a high degree of compatibility with the current ADS-B system. We acknowledge, however, that we have

not addressed the problem of establishing a means of message source integrity for ADS-B IN aircraft-to-aircraft messages. Future research into ADS-B message authentication and AA-MAC should work on developing source integrity solutions for ADS-B IN to mitigate the threat of message injection/modification attacks between airborne ADS-B senders and receivers.

Future work on security enhancements to the current ADS-B system could explore the security benefits of integrating AA-MAC for secure broadcast authentication with secure location verification techniques. The current Standard Terminal Automation Replacement System (STARS) air traffic control automation systems developed by Raytheon assimilates and filters data taken from surveillance radar, ADS-B and multilateration using Kalman filtering to verify target surveillance data [26]. Based on our findings, it would seem beneficial to pursue research into incorporating message authentication codes into these air traffic management automation systems. Combining the plausibility checks and track estimation provided by multilateration, data fusion and Kalman filtering with message authentication codes will greatly enhance the security of the existing ADS-B implementation.

Post-NextGen Future Research

Since there will eventually be a successor to ADS-B, it is also important for air traffic control system research to focus on the development of a secure surveillance protocol. Those responsible for creating a post-NextGen system should use the existing system as a case study and learn from the shortcomings inherent in the current ADS-B

implementation. Air traffic management communications signal integrity, security, resistance to jamming and a wider geographic coverage area are features that need to be incorporated into a post-NextGen ATM system.

The evolution of a post-NextGen system must have secure message authentication as a high priority. Taking into account the frequency capacity and message length constraints of the current ADS-B scheme, the most suitable cryptographic primitives need to be identified and developed into a post-NextGen system. In addition, an appropriate communication protocol that accommodates both air-to-air and air-to-ground messages must be identified. Equally important, a solution to the key management problem must be resolved [8]. A study of public key distribution and management for commercial aircraft in [13] defines a generic Airplane Asset Distribution System (AADS) and discusses several potential solutions for managing public keys that could be applied to a post-NextGen secure authentication scheme.

Adequate resistance to signal jamming must also be applied in a successor to ADS-B. Spread spectrum signal modulation provides protection against both jamming and DoS attacks, and should be considered in a follow-on air traffic management system design. Since FHSS is less susceptible to environmental interference factors than DSSS, it would seem that a frequency hopping approach would be more suitable for an air traffic management network. However, this will require additional frequency channel allocation for the network. Based on studies of L-band frequency allocation and frequency interference characteristics discussed in [27] and [28], it seems reasonable

that as the FAA moves to decommission some of its land-based NAVAIDS, there will be additional L-band frequencies that could be re-allocated for use in a future ATM system. These additional frequencies could be used in a FHSS scheme and would also mitigate concerns over increased congestion on the 1090 MHz band caused by the projected growth in air traffic.

An additional constraint of the current ADS-B implementation is the limited ADS-B system coverage available for aircraft operating routes that transit over remote locations of the globe. Aircraft operating on routes crossing bodies of water such as the Gulf of Mexico are in communication with ADS-B stations placed on oil drilling platforms, but no such solution is available for aircraft on long overwater oceanic routes.

Given the LOS range limitations of the 1090 MHz signal, solutions need to be developed for providing surveillance to aircraft operating in remote geographic locations. In [29], the authors explore the feasibility of utilizing the Iridium NEXT satellite constellation for use as ADS-B orbital stations. The authors study several cases that analyze the coverage rate available from an orbital-based system. The possibility of utilizing sub-orbital ADS-B stations is discussed in [30], where the authors experimented with stratospheric balloons to extend the ADS-B signal coverage range to over 300 NM as a proof-of-concept for ADS-B range extension. Development of a post-NextGen ATM system should seek to incorporate technologies that will permit accurate and timely surveillance of aircraft operating on routes that transit over remote geographic regions.

Conclusion

There is considerable current interest in providing security to NextGen and the ADS-B protocol. In this research, we discuss the existing work being done in ADS-B security and analyze cost-effective solutions to mitigating ADS-B vulnerabilities. NextGen and its GNSS-based surveillance component offer the potential to increase efficiency and capacity in the air traffic management system, but likely leave it more vulnerable to attack than the current radar-based surveillance system.

After reviewing the available alternatives, it is apparent that the solutions currently being researched can only be a compromise, providing a less than ideal improvement to the security of the present scheme. In order to implement comprehensive security into the ATM system, new message types and protocols need to be developed. The impact of increasing traffic density needs to be taken into account, and new protocols must be designed with scalability in mind in order to accommodate growth in ATM system communication network load. In planning for security solution in a post-NextGen ATM system, developers will need to incorporate both secure broadcast authentication and secure location verification in order to provide a more comprehensive security solution than those considered viable for the current surveillance system infrastructure.

REFERENCES

[1] A. Costin and A. Francillon, "Ghost in the Air(Traffic): On insecurity of ADS-B protocol and practical attacks on ADS-B devices," *Black Hat USA,* 2012.

[2] D. Magazu, "Exploiting the Automatic Dependent Surveillance-Broadcast system via false target injection," Air Force Institute of Technology, Wright-Patterson Air Force Base, Dayton, 2012.

[3] ICAO, "Status of ADS-B Avionics Equipage Along ATS Routes L642/M771 For Harmonized ADS-B Implementation.," in *ADS-B Seminar and 11th Meeting of ADS-B Study and Implementation Task Force*, Apr. 2012.

[4] Office of Inspector General, "ADS-B BENEFITS ARE LIMITED DUE TO A LACK OF ADVANCED CAPABILITIES AND DELAYS IN USER EQUIPAGE," U.S. Department of Transportation, Sep. 2014.

[5] C. J. Giannatto and G. Markowsky, "Potential Vulnerabilities of the NextGen Air Traffic Control System," in *World Congress in Computer Science, Computer Engineering and Applied Computing*, Las Vegas, Jul. 2014.

[6] M. Schäfer, V. Lenders and I. Martinovic, "Experimental Analysis of Attacks on Next Generation Air Traffic Communication," *Applied Cryptography and Network Security,* pp. 253-271, 2013.

[7] M. Strohmeier, V. Lenders and I. Martinovic, "On the Security of the Automatic Dependent Surveillance-Broadcast Protocol," *Communications Surveys & Tutorials,* vol. PP, no. 99, pp. 1-22, 2014.

[8] D. McCallie, J. Butts and R. Mills, "Security analysis of the ADS-B implementation in the next generation air transportation system," Air Force Institute of Technology, Wright-Patterson Air Force Base, Dayton, 2011.

[9] K. D. Wesson, T. E. Humphreys and B. L. Evans, "Can Cryptography Secure Next Generation Air Traffic Surveillance?," Tech. Rep., Jan. 2014.

[10] R. E. Boisvert and V. A. Orlando, "ADS-Mode S System Overview," MIT Lincoln Laboratory, Lexington, 1993.

[11] W. Z. Khan, M. Y. Aalsalem, M. N. Bin Mohammed Saad and Y. Xiang, "Detection and Mitigation of Node Replication Attacks in Wireless Sensor Networks: A Survey," *International Journal of Distributed Sensor Networks,* vol. 2013, pp. 1-22, 2013.

[12] N. Sastry, U. Shankar and D. Wagner, "Secure Verification of Location Claims," in *Proceedings of the 2nd ACM workshop on Wireless security*, San Diego, Sep. 2003.

[13] R. V. Robinson, M. Li, S. A. Lintelman, K. Sampigethaya, R. Poovendran, D. von Oheimb and J.-U. Bußer, "Impact of Public Key Enabled Applications on the Operation and Maintenance of Commercial Airplanes," in *AIAA Aviation Technology Integration, and Operations (ATIO) Conference*, Belfast, 2007.

[14] J. Zhang and V. Varadharajan, "Wireless sensor network key management survey and taxonomy," *Journal of Network and Computer Applications,* vol. 33, no. 2, pp. 63-75, Mar. 2010.

[15] A. Perrig, R. Canetti, J. D. Tygar and D. Song, "Efficient Authentication and Signing of Multicast Streams over Lossy Channels," in *2000 IEEE Symposium on Security and Privacy*, Oakland, May. 2000.

[16] A. Perrig, R. Canetti, J. D. Tygar and D. Song, "The TESLA Broadcast Authentication Protocol," *CryptoBytes,* vol. 5, no. 2, pp. 2-13, 2002.

[17] A. Perrig, R. Szewczyk, V. Wen, D. Culler and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks," *Wireless networks 8,* vol. 8, no. 5, pp. 521-534, 2002.

[18] Maximium Integrated Products, Inc., "An Introduction to Spread-Spectrum Communications," sorin-schwartz.com, [Online]. Available: http://sorin-schwartz.com/white_papers/fhvsds.pdf, Feb. 2003.

[19] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," in *SIGGRAPH 2001*, Los Angeles, Aug. 2001.

[20] J. Krozel, D. Andrisani, M. A. Ayoubi, T. Hozhizaki and C. Schwalm, "Aircraft ADS-B Data Integrity Check," *AIAA 4th Aviation Technology, Integration and Operations (ATIO) Forum,* pp. 1-11, 2004.

[21] N. Xu, R. Cassell, C. Evers, S. Hauswald and W. Langhans, "Performance assessment of Multilateration Systems - a solution to nextgen surveillance," in *Integrated Communications Navigation and Surveillance Conference (ICNS)*, Herndon, 2010.

[22] C. S. Hervaldo, W. B. Heinzelman, A. L. Murphy and C. J. N. Coelho, "A general data fusion architecture," *Information Fusion, 2003. Proceedings of the Sixth International Conference of,* vol. 2, pp. 1465-1472, Jul. 2003.

[23] T. Yong, W. Honggang, X. Zhili and H. Zhongtao, "ADS-B and SSR Data Fusion and Application," *Computer Science and Automation Engineering (CSAE), 2012 IEEE International Conference,* vol. 2, pp. 255-258, May 2012.

[24] S. Carey and A. Pasztor, "Report Faults Rollout of Air-Traffic-Control Upgrade," The Wall Street Journal, [Online]. Available: http://www.wsj.com/articles/report-faults-rollout-of-air-traffic-control-upgrade-1411518984, Sep. 2014.

[25] "FAA Aerospace Forecast Fiscal Years 2014-2034," Federal Aviation Administration, Oklahoma City, 2014.

[26] B. Kovell, B. Mellish, T. Newman and O. Kajopaiye, "Comparative Analysis of ADS-B Verification Techniques," University of Colorado, Boulder, May. 2012.

[27] M. Schnell, U. Epple, D. Shutin and N. Schneckenburger, "LDACS: Future Aeronautical Communications forAir-Traffic Management," *IEEE Communications Magazine,* pp. 104-110, May 2014.

[28] T. Otsuyama and S. Ozeki, "A Study of Evaluation Method for Aeronautical L-band Signal Environment during Flight Experiments," in *International Symposium on Electromagnetic Compatibility*, Gothenburg, 2014.

[29] P. Noschese, S. Porfili and S. Di Girolamo, "ADS-B via Iridium NEXT satellites," Thales Alenia Space Italia SpA, Rome, 2011.

[30] N. Demidovich, P. Purcell, R. Dewey, T. Lachenmeier, C. Greenlow, T. Willson, J. DiNofrio and D. Edwards, "Dual frequency ADS-B payload flight experiment on stratospheric balloon," in *Integrated Communications, Navigation and Surveillance Conference*, Herndon, 2014.

[31] G. Czerniak, "Kalman filter Single Variable Example," greg.czerniak.info, [Online] Available: http://greg.czerniak.info/guides/kalman1/kalman1.py.txt, 2014.

APPENDIX A: AA-MAC TEST PROGRAM

```
def parse_adsb(msg):
        '''
        Function to parse an ADS-B message into its main components. The Function
        converts the hexadecimal message into to its binary string equivalent and
        processes it into the 5 ADS-B message components.

        The function takes 1 parameter:
        @param msg: String value representing a legitimate 112-bit ADS-B data
        packet in hexadecimal format.
        @return: Returns a list of binary strings representing the ADS-B message
         components.
        '''
        msg = bin(int(msg,16))[2:]

        DF = ''
        CA = ''
        AA = ''
        Data = ''
        PI = ''
        for i in range(5):
         DF += msg[i]
        for i in range(5,8):
         CA += msg[i]
        for i in range(8,32):
         AA += msg[i]
        for i in range(32,88):
         Data += msg[i]
        for i in range(88,112):
         PI += msg[i]

        return [DF,CA,AA,Data,PI]

# End Function Code ------------------------------------------------------------
```

```
def create_block(hash, b_size=24):
       '''
       Function to create a hash of the designated block size using bit-wise XOR.

       Note that this implementation is designed to be a simple demonstration and is
       not intended to be computationally efficient.  There are certainly better
       data structures, such as mult-dimensional arrays, that would yield far more
       efficient algorithms than this iterative example.

       The function takes 1 required and 1 optional parameter:
    @param hash: String value representing a legitimate 112-bit ADS-B data
       packet in hexadecimal format.
    @param b_size: Integer value of the desired hash block size.  The default
       value is 24.
    @return Returns binary string of length b_size.
       '''

       # convert the hexadecimal hash value into its binary string equivalent
       hash = bin(int(hash,16))[2:]

       # setup the values used to iterate over the binary hash string
       h_size = len(hash)
       idx = 0
       b_hash = 0

       # iterate over the binary hash in b_size increments and XOR the substrings
       if h_size%b_size != 0:
        n = h_size/b_size
       else:
        n = h_size/b_size - 1
       for i in range(n):
        idx += b_size
        if idx < h_size - b_size:
               b_hash = int(hash[idx:idx+b_size],2) ^ b_hash
        else:
               b_hash = int(hash[idx:],2) ^ b_hash


       return bin(b_hash)[2:]


# End Function Code ------------------------------------------------------------
```

```
def test_hash(msg, key, b_size=24, n=1000, xor=True):
        '''
    Test function for detecting message authentication code failures, where the
    shortened hash is duplicated on distinctly different hash inputs.  The
    message simulates a message error by randomly changing one of the 112 bits
    in the ADS-B message.  The test compares the two hashes generated as the
    MAC to test for collisions due to the shortened hash.  The function
    prints out the number of collisions detected as a percentage of the total
    number of iterations.

    The function takes 2 required and 3 optional parameters:
    @param msg: Hexadecimal string representing a legitimate 112-bit ADS-B data
       packet.
    @param key: String value of the secret authentication key.
    @param b_size: Integer value of the desired hash block size.  The default
       value is 24.
       @param n: Integer value of the number of comparison iterations to perform.
       @param xor: Boolean value to determine if the create_block function is to
        be used to create the shortened MAC, or is a simple substring of the
        hash is to be used.
    '''

       import numpy as np
       from Crypto.Hash import MD5

       msg = parse_adsb(msg)[3]

       failure = 0
       for _ in range(n):
        # simulate error by modifying a random bit in msg
        i = np.random.randint(len(msg))
        if msg[i] == '0':
                msg_mod = msg[:i] + '1' + msg[i+1:]
        else:
                msg_mod = msg[:i] + '0' + msg[i+1:]
        h = MD5.new()
        h.update(msg+key)
        h_msg = h.hexdigest()
        h = MD5.new()
        h.update(msg_mod+key)
        h_msg_mod = h.hexdigest()

        if xor == True:
                test_msg = create_block(h_msg, b_size)
                test_msg_mod = create_block(h_msg_mod, b_size)
        else:
                test_msg = bin(int(h_msg,16))[2:b_size+2]
                test_msg_mod = bin(int(h_msg_mod,16))[2:b_size+2]

        if test_msg == test_msg_mod:
                print test_msg
                print test_msg_mod
                print '\n'
                failure += 1

       if failure != 0:
        print 'Failure percentage = {:.6f}%'.format(failure/float(n))
       else:
        print 'Test passed 100%'


# End Function Code ------------------------------------------------------------
```

APPENDIX B: SINGLE-VARIABLE KALMAN FILTER PROGRAM

```
import random
import numpy
import pylab


class kalman_linear(object):
    '''
    This class implements a linear Kalman filter and uses the parameter names as
    defined in [19].  This Kalman filter implementation is adapted from code
    written by Greg Czerniak [31].

    Instances of the kalman_linear class have 7 required parameters:

    @param A: Numpy matrix representing the state transition matrix.
    @param B: Numpy matrix representing the control matrix.
    @param H: Numpy matrix representing the observation matrix.
    @param x: Numpy matrix representing the initial state estimate.
    @param P: Numpy matrix representing the initial covariance estimate.
    @param Q: Numpy matrix representing the error in process estimate.
    @param R: Numpy matrix representing the error in measurement estimate.
    '''
    def __init__(self,A, B, H, x, P, Q, R):
        self.state_trans = A
        self.ctrl = B
        self.obs = H
        self.init_state_est = x
        self.init_cov_est = P
        self.proc_err_est = Q
        self.meas_err_est = R


# End __init__ Code ------------------------------------------------------------


    def get_state(self):
        '''
        Method to return the inital state estimate.

        @return: Returns a numpy matrix containing the initial state estimate
            value.
        '''
        return self.init_state_est


# End Method Code --------------------------------------------------------------
```

```
def step(self, ctrl_vec, meas_vec):
        '''
        Method to perform the Kalman filtering prediction, observation and update
        steps.
        '''
        # Prediction step
        pred_state_est = ( self.state_trans * self.init_state_est )\
                         + ( self.ctrl * ctrl_vec )
        pred_prob_est = ( (self.state_trans * self.init_cov_est )\
                        * numpy.transpose(self.state_trans ) )\
                        + self.proc_err_est

        # Observation step
        # residual - the discrepancy between the predicted and actual measurement
        residual = meas_vec - ( self.obs * pred_state_est )
        residual_cov = ( self.obs * pred_prob_est * numpy.transpose(self.obs) )\
                       + self.meas_err_est

        # Update step
        kalman_gain = pred_prob_est * numpy.transpose(self.obs)\
                      * numpy.linalg.inv(residual_cov)
        self.init_state_est = pred_state_est + ( kalman_gain * residual )

        # Determine the size of and create the identity matrix
        size = self.init_cov_est.shape[0]
        id_matrix = numpy.eye(size)

        # Update the covariance based on the results of the previous steps
        self.init_cov_est = (id_matrix - ( kalman_gain * self.obs ) )\
                            * pred_prob_est

# End Method Code ------------------------------------------------------------

# End kalman_linear Class definition -----------------------------------------

class noise_generator:
    '''
    This class implements a random noise generator for generating a noisy data
    series used to test a kalman_linear class instance.

    Instances of the noise_generator class require 2 parameters:

    @param mean: Float value of the mean for creating the Gaussian distribution.
    @param std_dev: Float value of the standard deviation used to create the
        Gaussian distribution.
    '''
    def __init__(self, mean, std_dev):
        self.mean = mean
        self.std_dev = std_dev


# End __init__ Code ----------------------------------------------------------


    def get_noise(self):
        '''
        Method to generate a random float value from a Gaussian distrubution.


        @return: Returns a random Float value from the generated Gaussian
            distribution.
        '''
        return random.gauss(self.mean, self.std_dev)


# End Method Code ------------------------------------------------------------
```

```
def get_mean(self):
        '''
        Method to return the mean of the Gaussian distribution.  This value
        represents the "actual" value that is being obscured by the noisy data.

        @return: Returns the float value passed in as the Gaussian distribution
            mean.
        '''
        return self.mean

# End Method Code -----------------------------------------------------------

# End noise_generator Class definition --------------------------------------


def test_kalman(A, B, H, x, P, Q, R, noise_mean, noise_std_dev, data_size):
    '''
    Test function for testing a kalman_linear class instance on a randomly
    generated noisy data series.

    The function generates a plot of the noisy data series, the estimated value
    returned by the linear Kalman filtering function and the actual value that
    the noisy data would be otherwise obscuring.

    The function takes 10 required parameters:

    @param A: Numpy matrix reperesenting the state transition matrix.
    @param B: Numpy matrix reperesenting the control matrix.
    @param H: Numpy matrix reperesenting the observation matrix.
    @param x: Numpy matrix representing the initial state estimate.
    @param P: Numpy matrix representing the initial covariance estimate.
    @param Q: Numpy matrix representing the error in process estimate.
    @param R: Numpy matrix representing the error in measurement estimate.
    @param mean: Float value of the mean for creating the Gaussian distribution.
    @param std_dev: Float value of the standard deviation used to create the
        Gaussian distribution.
    @param data_size: Integer value indicating the desired size of the test
        data series.
    '''

    # Create class instances
    filter = kalman_linear(A,B,H,xhat,P,Q,R)
    generator = noise_generator(noise_mean, noise_std_dev)

    # Create storage for generated data
    noisy_data = []
    actual_val = []
    kalman = []

    # Create the noisy and filtered linear data series
    for i in range(data_size):
        noisy_val = generator.get_noise()
        noisy_data.append(noisy_val)
        actual_val.append(generator.get_mean())
        kalman.append(filter.get_state()[0,0])
        filter.step(numpy.matrix([0]), numpy.matrix([noisy_val]))


    # Create plot of the noisy and filtered data
    pylab.plot(range(data_size), noisy_data, 'r', range(data_size),
               kalman, 'g', range(data_size), actual_val, 'b')
    pylab.xlabel('Time')
    pylab.ylabel('Noisy Data')
    pylab.title('Estimation of Noisy Data Series with Kalman Filter')
    pylab.legend(('Noisy Values','Kalman Filtered Values', 'Actual Value'))
    pylab.show()


# End Function Code ---------------------------------------------------------
```

BIOGRAPHY OF THE AUTHOR

Carl J. Giannatto, Jr. was born in Jersey City, NJ on May 7, 1964 and graduated from Satellite High School (Florida) in 1982. In December 1988, he received a BA degree in Finance from the University of South Florida (USF) in Tampa, Florida. Following graduation, he attended Aviation Officer Candidate School at Pensacola Naval Air Station and was commissioned an Ensign in the Naval Reserve in July 1990.

He began his aviation career at Naval Air Station Whiting Field, and upon completion of flight training was assigned to Patrol Squadron Twenty-Six at Naval Air Station Brunswick, Maine. Upon leaving the Naval Reserve, he transferred to the Maine Air National Guard and joined the 101st Air Refueling Wing in Bangor, Maine. During this same period, he also began flying as a commercial airline pilot. During his flying career, he has accumulated over 10,000 hours of combined military, commercial and general aviation flying experience in a wide variety of aircraft. He has served as a ground school, simulator and flight instructor, as well as an FAA-designated line check pilot in the commercial airline industry. He is currently an Aircraft Commander in the Maine Air National Guard with the rank of LtCol and a First Officer for American Airlines.

Carl J. Giannatto, Jr. is a candidate for the Master of Science degree in Computer Science from The University of Maine in May 2015.